AD 689821

TM-738/050/00

SCIENTIFIC REPORT
NO. 23

MULTITAPE AFA

Shelia Greibach
and Seymour Ginsberg

2 January 1969

Prepared for
AIR FORCE CAMBRIDGE RESEARCH LABORATORIES
OFFICE OF AEROSPACE RESEARCH, USAF
BEDFORD, MASSACHUSETTS
and
AIR FORCE OFFICE OF SCIENTIFIC RESEARCH
OFFICE OF AEROSPACE RESEARCH, USAF
ARLINGTON, VIRGINIA

Contract Monitor: Thomas V. Griffiths – Data Sciences Laboratory
Monitored by: R. Swanson, SRIR (AFOSR) Supported by
SDC'S Independent Research Program:
Contract F1962867C0008, Programming (Algorithmic) Languages,
Project No. 5632. Task No. 563205; and Grant No. AF-AFOSR-1203-67
Work Unit No. 56320501

AFCRL-69-0173

TM-738/050/00

SDC TM SERIES

The work reported herein was supported by SDC and Contract F1962867C0008, Programming (Algorithmic) Languages; Project No. 5632, Task No. 563205; Work Unit No. 56320501; and Grant No. AF-AFOSR-1203-67

SCIENTIFIC REPORT NO. 23

2 January 1969

Multitape AFA

by

Sheila Greibach*

and

Seymour Ginsburg**

Monitored by: Contract Monitor
Thomas V. Griffiths,
Data Sciences Laboratory
R. Swanson, SRIR (AFOSR)

Prepared for

AIR FORCE CAMBRIDGE RESEARCH LABORATORIES
OFFICE OF AEROSPACE RESEARCH, USAF
BEDFORD, MASSACHUSETTS 01730
and
AIR FORCE OFFICE OF SCIENTIFIC RESEARCH
OFFICE OF AEROSPACE RESEARCH, USAF
ARLINGTON, VIRGINIA

SYSTEM
DEVELOPMENT
CORPORATION
2500 COLORADO AVE.
SANTA MONICA
CALIFORNIA
90406

*Harvard University and System Development Corporation.

**System Development Corporation and University of Southern California

ABSTRACT

The present paper gives device representations, via multitape AFA, for the families of languages which result from applying the $\wedge$ and the substitution operations to AFL. In particular, if $\mathcal{D}_1$ and $\mathcal{D}_2$ are multitape AFA (i.e., certain families of multi-storage tape acceptors), then $\mathcal{D}_1 \wedge \mathcal{D}_2$ is defined as the family of multitape acceptors which results when the tapes of $\mathcal{D}_1$ and $\mathcal{D}_2$ are coalesced, with the $\mathcal{D}_1$- tapes preceding those in $\mathcal{D}_2$. It is shown that the smallest full AFL containing $\mathcal{L}(\mathcal{D}_1) \wedge \mathcal{L}(\mathcal{D}_2) = \{L_1 \cap L_2 / L_1 \text{ in } \mathcal{L}(\mathcal{D}_1)\}$ is $\mathcal{L}(\mathcal{D}_1 \wedge \mathcal{D}_2)$. For each multitape AFA $\mathcal{D}$, a set $\mathcal{D}^N$ of "nested" multitape acceptors is defined. It is shown that if $\mathcal{D}_1$ and $\mathcal{D}_2$ are single-tape AFA, then the family of languages obtained from $(\mathcal{D}_1 \wedge \mathcal{D}_2)^N$ is the family of languages obtained by substituting the AFL defined by $\mathcal{D}_2$ into the AFL defined by $\mathcal{D}_1$.

MULTITAPE AFA*

INTRODUCTION

In [5] and [18], the notion of a family of one-way nondeterministic devices was abstracted and studied extensively. A natural extension of a device with a particular type of storage tape is a multitape (storage) device, each tape of the same kind. For example, a pushdown acceptor can be extended to a device with two pushdown tapes. In most familiar cases--counter, pushdown, stack--adding a second storage tape increases the power of the device to that of a Turing acceptor. By suitable restrictions on multitape devices, families can be obtained so that the associated languages, as, for example, the one-way nondeterministic list-storage languages [8], do not include all recursively enumerable sets. The purpose of this paper is to abstract the notion of an "abstract family of multitape acceptors" (abbreviated "multitape AFA"), each storage tape not necessarily of the same kind, and examine the family of associated languages.

Our interest in multitape AFA originally arose from studying various operations upon families of languages. We were interested in the operations of $\wedge$ and substitution among families of languages, these operations appearing, sometimes in disguised form, in a number of papers [7, 8, 11, 12, 13, 15, 16, 20, 21]. (If $\mathcal{L}_1$ and $\mathcal{L}_2$ are families of languages, then $\mathcal{L}_1 \wedge \mathcal{L}_2$ =

*Research sponsored in part by the Air Force Cambridge Research Laboratories, Office of Aerospace Research, USAF, under contract F1962867C0008, and by the Air Force Office of Scientific Research, Office of Aerospace Research, USAF, under AFOSR Grant No. AF-AFOSR-1203-67A.

$\{L_1 \cap L_2 / \text{each } L_1 \text{ in } \mathcal{L}_1\}$.) Now certain families of languages are representable by single-tape AFA (see [5]). For these families, we were interested in device representations for the families obtained by the $\wedge$ and substitution operations. The main results of this document show that multitape AFA provide such representations.

The paper itself is divided into four sections and an appendix. Section 1 introduces the notion of a multitape AFA. It is shown (Lemma 1.1) that for each multitape AFA there exists a single-tape AFA, equivalent from the point of view of sets accepted.

Section 2 introduces the operation of $\wedge$ between multitape AFA and discusses the operation of $\wedge$ between AFL. (AFL [5] are families of sets of words with certain properties and are an abstraction of many of the formal languages discussed in computer science.) Roughly speaking, if $\mathcal{D}_1$ and $\mathcal{D}_2$ are multitape AFA, then $\mathcal{D}_1 \wedge \mathcal{D}_2$ is the multitape AFA resulting when the tapes of $\mathcal{D}_1$ and $\mathcal{D}_2$ are coalesced, with the $\mathcal{D}_1$-tapes preceding those in $\mathcal{D}_2$. The operation $\wedge$ between multitape AFA is then used to provide a multitape AFA characterization of the smallest AFL containing $\mathcal{L}_1 \wedge \ldots \wedge \mathcal{L}_n$, each $\mathcal{L}_i$ an AFL, in terms of an AFA defining the $\mathcal{L}_i$ (Theorem 2.1). A characterization of an AFL being closed under intersection is given in terms of the existence of a certain kind of multitape AFA (Theorem 2.3).

Section 3 is concerned with multitape transducers, i.e., devices obtained by adding an output tape to a multitape acceptor. Connections between multitape transducers, composition of single-tape transducers, and $\wedge$ are then found.

Section 4 deals with "nested" multitape AFA. These are collections of multitape acceptors, each acceptor changing at most one storage tape at a time, and all tapes to the right of the changed one empty. The main result (Theorem 4.2) here is that if $\mathcal{D}_1$ and $\mathcal{D}_2$ are single-tape AFA then the AFL given by the nested acceptors of $\mathcal{D}_1 \wedge \mathcal{D}_2$ is the family obtained by substituting the AFL defined by $\mathcal{D}_2$ into the AFL defined by $\mathcal{D}_1$. In demonstrating this result, a technical lemma (Lemma 4.3) is used whose proof is so extensive that it is relegated to an appendix.

Numerous applications of the theory are given throughout to AFL and AFA of interest in computer science. For example, it is shown in Section 2 that a language (L) can be recognized in quasi-realtime by a multitape Turing acceptor if and only if L can be recognized in quasi-realtime by a multi-pushdown tape acceptor if and only if L is the ϵ-free homomorphic image of the finite intersection of context-free languages. The applications given show that multitape AFA provide greater insight into families of languages of concern to automata and formal language theorists.

Section 1. Preliminaries

As mentioned in the introduction, our aim in this paper is to study multitape (storage) devices, each tape not necessarily of the same kind. In addition, we shall examine the families of languages associated with these

devices. In this section we formalize the kind of multitape device with which we are concerned, and present some examples.

In [5] we formulated the notion of an AFA (abstract family of acceptors) and established a basic connection between it and certain families of languages. We shall define the multitape acceptors of interest to us by modifying the notion of an AFA. In particular, we introduce the notion of an "AFA-schema," a construct which represents a type of auxiliary storage tape. We then define a multitape AFA as a family of devices, each of which has only a finite number of preassigned AFA-schema.

<u>Definition</u>. An <u>AFA-schema</u> is a 4 tuple (Γ, I, f, g), with the following properties:

(a) Γ and I are abstract sets, with Γ and I nonempty.

(b) f is a function from[1] $\Gamma^* \times I$ into $\Gamma^* \cup \{\emptyset\}$.

(c) g is a function from Γ^* into the finite subsets of Γ^* such that $g(\epsilon) = \{\epsilon\}$, and ϵ is in $g(\gamma)$ if and only if $\gamma = \epsilon$.

(d) For each γ in $g(\Gamma^*)$[2], there is an element 1_γ in I satisfying $f(\gamma', 1_\gamma) = \gamma'$ for all γ' such that $g(\gamma')$ contains γ.

(e) For each u in I, there exists a finite set $\Gamma_u \subseteq \Gamma$ with the following property: If $\Gamma_1 \subseteq \Gamma$, γ is in Γ_1^*, and $f(\gamma, u) \neq \emptyset$, then $f(\gamma, u)$ is in $(\Gamma_1 \cup \Gamma_u)^*$; that is, for each γ in Γ^*, each symbol occurring in $f(\gamma, u)$ either occurs in γ or is in Γ_u.

---

(1) For each abstract set E, E^* is the set of all **finite** strings of symbols from E, including the empty string ϵ. Each element of E^* is called a <u>word</u> in E.

(2) For each set A, $g(A) = \bigcup_{\gamma \text{ in } A} g(\gamma)$.

Intuitively, an AFA-schema is a type of auxiliary storage. Γ is the set of "auxiliary" symbols (i.e., the set of symbols going into the auxiliary storage). I is the set of "instructions," g is the "storage" information function which interprets the auxiliary storage configuration, and f is the "storage transformation" function which produces a new auxiliary storage configuration. The reader is referred to [5] for further details.

Definition. A multitape AFA is an ordered pair $(\Omega,\mathfrak{D})$, or $\mathfrak{D}$ when Ω is understood, with the following properties:

(1) Ω is a 5-tuple $(K,\Sigma,\mathcal{A},<,\mu)$, where

(a) $\mathcal{A}$ is a nonempty index set and $<$ is a simple order on $\mathcal{A}$[3].

(b) μ is a function on $\mathcal{A}$ such that for each α in $\mathcal{A}$, $\mu(\alpha) = X_\alpha = (\Gamma_\alpha, I_\alpha, f_\alpha, g_\alpha)$ is an AFA-schema.[4]

(c) K and Σ are infinite abstract sets.

(2) $\mathfrak{D}$ is the family of all 6-tuples $D = (K_1,\Sigma_1,\delta,q_0,F,\upsilon)$, called multitape acceptors, where

(a) $\upsilon = (\alpha_1, \ldots, \alpha_k)$, k finite, α_i in $\mathcal{A}$ for each i, and $\alpha_i < \alpha_{i+1}$ for $1 \leq i < k$.

(b) K_1 and Σ_1 are finite subsets of K and Σ, resp., F is a subset of K_1, and q_0 is in K_1.

---

(3) That is, $<$ is transitive, antireflexive, and dichotomous.

(4) Thus the component μ could be replaced in Ω by the more cumbersome symbolism $\{X_\alpha\}_{\alpha \text{ in } \mathcal{A}}$.

(c) δ is a function from $K_1 \times (\Sigma_1 \cup \{\epsilon\}) \times (\Gamma_{\alpha_1} \times \ldots \times \Gamma_{\alpha_k})$ into the finite subsets of $K_1 \times (I_{\alpha_1} \times \ldots \times I_{\alpha_k})$ such that

$$G_D = \{(\gamma_1, \ldots, \gamma_k)/\delta(q, a, (\gamma_1, \ldots, \gamma_k)) \neq \emptyset \text{ for some } q \text{ and } a\}$$

is finite.

As in an AFA, K is the set of all possible "states" and Σ is the set of all possible inputs. $\mathfrak{A}$ is an index set and μ assigns a type of auxiliary storage to each α in $\mathfrak{A}$. Each multitape acceptor has a finite number of tapes, with μ and ν indicating their types. The order in which a device displays its tapes does not really affect its action. The role of $<$ is just to impose some order.

In general, for each α in $\mathfrak{A}$ and each γ in $g_\alpha(\Gamma_\alpha^*)$, there may be more than one element $u_{\alpha,\gamma}$ (possibly an infinite number) in I_α satisfying $f_\alpha(\gamma', u_{\alpha,\gamma}) = \gamma'$ for all γ' such that $g_\alpha(\gamma')$ contains γ. Now we shall frequently be defining acceptors with special properties. Since acceptors are finitely described, we shall need a specific such $u_{\alpha,\gamma}$ for each α and each γ. Hence we have the following.

<u>Notation</u>. For each α in $\mathfrak{A}$ and each γ in $g_\alpha(\Gamma_\alpha^*)$, $1(\alpha,\gamma)$ denotes a specific element in I_α satisfying $f_\alpha(\gamma', 1(\alpha,\gamma)) = \gamma'$ for all γ' such that $g_\alpha(\gamma')$ contains γ. In case $\gamma = \epsilon$, $1(\alpha,\epsilon)$ is abbreviated 1_α.

The movement of a multitape acceptor is now described, in analogy with that of an acceptor in an AFA.

<u>Definition</u>. Let $D = (K_1, \Sigma_1, \delta, q_0, F, \nu)$, $\nu = (\alpha_1, \ldots, \alpha_k)$, be a multitape acceptor. A <u>configuration</u> C is a (k+2)-tuple $C = (q, w, (\gamma_1, \ldots, \gamma_k))$, where q is in K_1, w is in Σ_1^*, and each γ_i is in $\Gamma_{\alpha_i}^*$.

<u>Notation</u>. Let $\vdash$ be the relation on configurations defined as follows: For a in $\Sigma_1 \cup \{\epsilon\}$ and w in Σ_1^*, $(q, aw, (\gamma_1, \ldots, \gamma_k)) \vdash (q', w, (\gamma_1', \ldots, \gamma_k'))$ if there exist $\bar{\gamma}_1, \ldots, \bar{\gamma}_k$, each $\bar{\gamma}_i$ in $g_{\alpha_i}(\gamma_i)$, such that $(q', (u_1, \ldots, u_k))$ is in $\delta(q, a, (\bar{\gamma}_1, \ldots, \bar{\gamma}_k))$ and $f_{\alpha_i}(\gamma_i, u_i) = \gamma_i'$ for each i. For each $i \geq 0$ let $\overset{i}{\vdash}$ be the relation on configurations defined by induction as follows: $C \overset{0}{\vdash} C$ for each C and $C \overset{n+1}{\vdash} C'$ if there exists C'' such that $C \overset{n}{\vdash} C''$ and $C'' \vdash C'$. Let $\overset{*}{\vdash}$ be the transitive, reflexive extension of $\vdash$, i.e., $C \overset{*}{\vdash} C'$ if $C \overset{n}{\vdash} C'$ for some $n \geq 0$.

As usual, the above relations are written $\underset{D}{\vdash}$, $\overset{i}{\underset{D}{\vdash}}$, and $\overset{*}{\underset{D}{\vdash}}$ if D is to be emphasized.

For each multitape AFA $\mathcal{D}$ and each $\upsilon = (\alpha_1, \ldots, \alpha_k)$, $k \geq 1$, we shall be interested in those multitape acceptors with auxiliary storage tapes υ. Hence we have

<u>Notation</u>. For each multitape AFA $\mathcal{D}$ and each υ, let $\mathcal{D}_\upsilon$ be the family of all $D = (K_1, \Sigma_1, \delta, q_0, F, \upsilon)$ in $\mathcal{D}$.

We now introduce the families of sets of words defined by the previous families of acceptors.

<u>Notation</u>. For each $D = (K_1, \Sigma_1, \delta, q_0, F, \upsilon)$ in $\mathcal{D}$, let

$$L(D) = \{w/(q_0, w, (\epsilon, \ldots, \epsilon)) \overset{*}{\underset{D}{\vdash}} (q, \epsilon, (\epsilon, \ldots, \epsilon)) \text{ for some } q \text{ in } F\}.$$

Let $\mathcal{L}(\mathcal{D}) = \{L(D)/D \text{ in } \mathcal{D}\}$ and for each υ, $\mathcal{L}(\mathcal{D}_\upsilon) = \{L(D)/D \text{ in } \mathcal{D}_\upsilon\}$.

As in the acceptors discussed in [5], so we frequently shall be interested in those multitape acceptors with a bounded number of consecutive ϵ-moves.

<u>Definition</u>. D in $\mathcal{D}$ is <u>quasi-realtime</u> if there exists an integer $m \geq 0$ such that for all configurations $C = (q, \epsilon, (\gamma_1, \ldots, \gamma_k))$ and $C' = (q', \epsilon, (\gamma_1', \ldots, \gamma_k'))$,

$C \overset{n}{\vdash} C'$ implies $n \leq m$. Let

$\mathcal{L}^t(\mathcal{D}_\nu) = \{L(D)/D \text{ in } \mathcal{D}_\nu \text{ is quasi-realtime}\}$

and $\mathcal{L}^t(\mathcal{D}) = \{L(D)/D \text{ in } \mathcal{D} \text{ is quasi-realtime}\}$.

If $\nu_1 = (\alpha_1, \ldots, \alpha_k)$, $\nu_2 = (\beta_1, \ldots, \beta_\ell)$, $A = \{\alpha_1, \ldots, \alpha_k\}$, $B = \{\beta_1, \ldots, \beta_\ell\}$ and $A \subseteq B$, then $\mathcal{D}_{\nu_1}$ can be "embedded" in $\mathcal{D}_{\nu_2}$ in the following sense: Each element D of $\mathcal{D}_{\nu_1}$ may be identified with the multitape acceptor D'(in $\mathcal{D}_{\nu_2}$)which consists of (i) the tapes of D, and (ii) the tapes in B-A, with ϵ on them, acting under the identity instruction. (D' behaves "essentially" the same as D, is quasi-realtime if and only if D is, and is such that $L(D') = L(D)$.) Thus $\mathcal{L}(\mathcal{D}_{\nu_1}) \subseteq \mathcal{L}(\mathcal{D}_{\nu_2})$ and $\mathcal{L}^t(\mathcal{D}_{\nu_1}) \subseteq \mathcal{L}^t(\mathcal{D}_{\nu_2})$.

The most important multitape AFA are those having just one tape.

<u>Definition</u>. If $\mathfrak{A} = \{\alpha\}$, then $(\Omega,\mathcal{D})$ is said to be a <u>single-tape</u> AFA.

We may identify each single-tape AFA with an AFA as defined in [5]. In particular, if $(\Omega,\mathcal{D})$ is a single-tape AFA, then we may regard Ω as $(K,\Sigma,\Gamma_\alpha,I_\alpha,f_\alpha,g_\alpha)$ and $\mathcal{D}$ as the set of all D of the form $(K_1,\Sigma_1,\delta,p_0,F)$.

If $(\Omega,\mathcal{D})$ is a finite-tape AFA, i.e., $\mathfrak{A} = \{\alpha_1, \ldots, \alpha_k\}$ for some finite k, with $\alpha_i < \alpha_{i+1}$ for each i, then as noted above, $\mathcal{D}$ may be identified with $\mathcal{D}_\nu$, $\nu = (\alpha_1, \ldots, \alpha_k)$.

We first show that multitape AFA are equivalent (from the point of view of sets accepted) to single-tape AFA. Thus, as families of recognition devices, multitape AFA are no more powerful than single-tape AFA. However, as we shall see, multitape acceptors are useful in representing, in a "natural" way, the languages obtained from families of languages by certain operations.

<u>Lemma 1.1</u>. If $(\Omega,\mathcal{D})$ is a multitape AFA, then there exists a single-tape AFA

$(\bar{\Omega},\bar{\mathscr{S}})$ such that $\mathcal{L}(\bar{\mathscr{S}})= \mathcal{L}(\mathscr{S})$ and $\mathcal{L}^t(\bar{\mathscr{S}})= \mathcal{L}^t(\mathscr{S})$.

Proof. Let $\Omega = (K,\Sigma,\mathcal{Q},<,\mu)$ and let ξ be a new symbol not in $\bigcup_{\alpha \text{ in } \Omega} \Gamma_\alpha$. For each $\nu = (\alpha_1, \ldots, \alpha_k)$ and each $(u_1, \ldots, u_k)$ in $I_{\alpha_1} \times \ldots \times I_{\alpha_k}$, let $\sigma(\nu)$ and $\sigma(\nu,(u_1, \ldots, u_k))$ be new symbols. Let $\bar{\Gamma} = \bigcup_{\alpha \text{ in } \mathcal{Q}} \Gamma_\alpha \cup \{\xi\} \cup \{\sigma(\nu)/\text{all } \nu\}$ and

$$\bar{I} = \{\sigma(\nu,(u_1, \ldots, u_k))/\text{all } \sigma(\nu,(u_1, \ldots, u_k))\} \cup \{\sigma(\nu)/\text{all } \nu\} \cup \{\epsilon\}.$$

For $(\gamma_1, \ldots, \gamma_k)$ in $\Gamma^*_{\alpha_1} \times \ldots \times \Gamma^*_{\alpha_k}$ and $\nu = (\alpha_1, \ldots, \alpha_k)$, let

$$\bar{g}(\sigma(\nu)\xi\gamma_1\xi \ldots \gamma_k\xi) = \sigma(\nu)\xi g_{\alpha_1}(\gamma_1)\xi \ldots g_{\alpha_k}(\gamma_k)\xi.$$

Let $\bar{g}(\epsilon) = \{\epsilon\}$. For $\nu = (\alpha_1, \ldots, \alpha_k)$, $(u_1, \ldots, u_k)$ in $I_{\alpha_1} \times \ldots \times I_{\alpha_k}$, $(x_1, \ldots, x_k)$ in $\Gamma^*_{\alpha_1} \times \ldots \times \Gamma^*_{\alpha_k}$, and $\bar{x}_i = f_{\alpha_i}(x_i,u_i)$, let $\bar{f}(\epsilon,\sigma(\nu)) = \sigma(\nu)\xi^{k+1}$, $\bar{f}(\sigma(\nu)\xi^{k+1},\epsilon) = \epsilon$, $\bar{f}(\epsilon,\epsilon) = \epsilon$, and

$$\bar{f}(\sigma(\nu)\xi x_1\xi \ldots x_k\xi, \sigma(\nu,(u_1, \ldots, u_k))) = \sigma(\nu)\xi\bar{x}_1\xi \ldots \bar{x}_k\xi.^{(5)}$$

We first show that for $\bar{\Omega} = (K,\Sigma,\bar{\Gamma},\bar{I},\bar{f},\bar{g})$, $(\bar{\Omega},\bar{\mathscr{S}})$ is a single-tape AFA. Then we consider $\mathcal{L}(\bar{\mathscr{S}})$ and $\mathcal{L}^t(\bar{\mathscr{S}})$.

Intuitively, $\bar{\Omega}$ is the single-tape AFA obtained by taking the tapes of each finite set of tapes and placing them on one tape, in the obvious order, with appropriate separators. Formally, we first note the following (for $\nu = (\alpha_1, \ldots, \alpha_k)$):

(1) $\bar{f}(\sigma(\nu)\xi x_1 \ldots \xi x_k\xi, \sigma(\nu,(1(\alpha_1,\gamma_1),\ldots , 1(\alpha_k,\gamma_k))))= \sigma(\nu)\xi x_1 \ldots\xi x_k\xi$ for all $\sigma(\nu)\xi x_1 \ldots \xi x_k\xi$ such that $\bar{g}(\sigma(\nu)\xi x_1 \ldots \xi x_k\xi)$ contains $\sigma(\nu)\xi\gamma_1 \ldots\xi\gamma_k\xi$.

(2) For all x, $\bar{f}(x,\sigma(\nu))$ is in $\{\sigma(\nu),\xi\}^* \cup\{\emptyset\}$ and $\bar{f}(x,\epsilon)$ is in $\{\epsilon,\emptyset\}$.

For each $(u_1, \ldots, u_k)$ in $I_{\alpha_1} \times \ldots \times I_{\alpha_k}$ and $\Gamma_{u_1}, \ldots, \Gamma_{u_k}$ as in (e) of the definition of an AFA-schema, $\Gamma_{\sigma(\nu,(u_1, \ldots, u_k))} = \bigcup_k \Gamma_{u_i} \cup\{\sigma(\nu),\xi\}$.

---

(5) Functional values are always to be $\emptyset$ unless otherwise stated.

Hence $(\bar{\Gamma}, \bar{I}, \bar{f}, \bar{g})$ is an AFA-schema.

Suppose that $D = (K_1, \Sigma_1, \delta, q_o, F, \upsilon)$ is an arbitrary multitape acceptor in $\mathcal{D}$. Let $\bar{q}_o$ and r_o be new symbols in K and $\bar{D} = (K_1 \cup \{\bar{q}_o, r_o\}, \Sigma_1, \bar{\delta}, \bar{q}_o, \{r_o\})$, where $\bar{\delta}$ is defined as follows:

(3) $(q', \sigma(\upsilon, (u_1, \ldots, u_k)))$ is in $\bar{\delta}(q, a, \sigma(\upsilon)\xi\gamma_1 \ldots \xi\gamma_k\xi)$ if $(q', (u_1, \ldots, u_k))$ is in $\delta(q, a, (\gamma_1, \ldots, \gamma_k))$.

(4) (r_o, ϵ) is in $\bar{\delta}(p, \epsilon, \sigma(\upsilon)\xi^{k+1})$ if p is in F.

(5) $\bar{\delta}(\bar{q}_o, \epsilon, \epsilon) = \{(q_o, \sigma(\upsilon))\}$.

Clearly $L(D) = L(\bar{D})$ and $\bar{D}$ is quasi-realtime if D is quasi-realtime. Thus $\mathcal{L}(\mathcal{D}) \subseteq \mathcal{L}(\bar{\mathcal{D}})$ and $\mathcal{L}^t(\mathcal{D}) \subseteq \mathcal{L}^t(\bar{\mathcal{D}})$.

Clearly $L(D) = L(\bar{D})$ and $\bar{D}$ is quasi-realtime. Thus $\mathcal{L}(\mathcal{D}) \subseteq \mathcal{L}(\bar{\mathcal{D}})$ and $\mathcal{L}^t(\mathcal{D}) \subseteq \mathcal{L}^t(\bar{\mathcal{D}})$.

Now let $\bar{D} = (K_1, \Sigma_1, \delta, q_o, F)$ be in $\bar{\mathcal{D}}$. Note that many different $\sigma(\upsilon)$ might appear in $\bar{D}$. Let

$S = \{\upsilon / (q', \sigma(\upsilon))$ in $\delta(q, a, \epsilon)$ for some q and a$\}$.

We may assume that if $(q', \sigma(\upsilon_1)\xi u_1\xi \ldots u_k\xi)$ is in $\delta(q, a, \sigma(\upsilon_2)\xi\gamma_1 \ldots \xi\gamma_k\xi)$, then $\upsilon_1 = \upsilon_2 = (\alpha_1, \ldots, \alpha_k)$, where u_i is in I_{α_i} and γ_i is in $g_{\alpha_i}(\Gamma^*_{\alpha_i})$, $1 \le i \le k$, and υ_1 is in S. (For no other type of rule can be applied in the $\vdash$ relation.) Since S is finite, there exists some $\upsilon_o = (\alpha_1, \ldots, \alpha_n)$ such that if υ is in S, then $\upsilon = (\alpha_{j_1}, \ldots, \alpha_{j_k})$ for some $1 \le j_1 < \ldots < j_k \le n$. Let $K' = K_1 \times (S \cup \{\epsilon\})$. Let $D' = (K', \Sigma_1, \delta', (q_o, \epsilon), F \times \{\epsilon\}, \upsilon_o)$, where δ' is defined as follows (for arbitrary $\upsilon = (\alpha_{j_1}, \ldots, \alpha_{j_k})$):

(6) $((q', \epsilon), (1_{\alpha_1}, \ldots, 1_{\alpha_n}))$ is in $\delta'((q, \epsilon), a, (\epsilon, \ldots, \epsilon))$ if (q', ϵ) is in $\delta(q, a, \epsilon)$.

(7) $((q'\epsilon), (1_{\alpha_1}, \ldots, 1_{\alpha_n}))$ is in $\delta'((q, \upsilon), a, (\epsilon, \ldots, \epsilon))$ if $(q'\epsilon)$ is in $\delta(q, a, \sigma(\upsilon)\xi^{k+1})$.

(8) $((q',\nu),(1_{\alpha_1}, \ldots, 1_{\alpha_n}))$ is in $\delta'((q,\epsilon),a,(\epsilon, \ldots, \epsilon))$ if $(q',\sigma(\nu))$ is in $\delta(q,a,\epsilon)$.

(9) $((q',\nu),(u_1', \ldots, u_n'))$ is in $\delta'((q,\nu),a,(\gamma_1', \ldots, \gamma_n'))$ if $(q',\sigma(\nu,(u_1, \ldots, u_k)))$ is in $\delta(q,a,\sigma(\nu)\xi\gamma_1 \ldots \xi\gamma_k\xi)$ where γ_i in $\Gamma_{\alpha_{j_i}}^*$ for each i, $u_i' = u_i$ and $\gamma_i' = \gamma_i$ for i in $\{j_1, \ldots, j_k\}$, and $u_i' = 1_{\alpha_i}$ and $\gamma_i' = \epsilon$ for i not in $\{j_1, \ldots, j_k\}$.

It is readily seen that $L(D') = L(\bar{D})$ and D' is quasi-realtime if $\bar{D}$ is quasi-realtime. Thus $\mathcal{L}(\mathfrak{D}) = \mathcal{L}(\bar{\mathfrak{D}})$ and $\mathcal{L}^t(\mathfrak{D}) \subseteq \mathcal{L}^t(\bar{\mathfrak{D}})$, whence equality in both cases.

Since each $\mathfrak{D}_\nu$ may be considered a multitape AFA, we have

Corollary. If $(\Omega,\mathfrak{D})$ is a multitape AFA, then for each ν there exists a single-tape AFA $(\bar{\Omega},\bar{\mathfrak{D}})$ such that $\mathcal{L}(\bar{\mathfrak{D}}) = \mathcal{L}(\mathfrak{D}_\nu)$ and $\mathcal{L}^t(\bar{\mathfrak{D}}) = \mathcal{L}^t(\mathfrak{D}_\nu)$.

Using Lemma 1.1, we are able to apply results of [5] to multitape AFA.

Since we are dealing with acceptors, we are naturally interested in various families of sets of words. We recall some terminology from [7, 5].

Definition. A family of languages is pair $(\Sigma,\mathcal{L})$, or $\mathcal{L}$ when Σ is understood, where

(1) Σ is an infinite set of symbols,

(2) for each L in $\mathcal{L}$ there is a finite set $\Sigma_1 \subseteq \Sigma$ such that $L \subseteq \Sigma_1^*$, and

(3) $L \neq 0$ for some L in $\mathcal{L}$.

The families of languages we are most concerned with are now given.

Definition. An abstract family of languages (abbreviated AFL) is a family

of languages closed under the operations of $\cup$, $\cdot$, $^+$,[6] ϵ-free homomorphism[7], inverse homomorphism, and intersection with regular sets. An AFL is said to be <u>full</u> if it is closed under arbitrary homomorphism.

It was shown in [5] that for each single-tape AFA $\mathcal{D}$, $\mathcal{L}^t(\mathcal{D})$ and $\mathcal{L}$ are AFL containing $\{\epsilon\}$, with $\mathcal{L}(\mathcal{D})$ being full (and conversely, for each AFL $\mathcal{L}$ containing $\{\epsilon\}$, resp. full AFL $\mathcal{L}$, there exists a single-tape AFA $\mathcal{D}$ such that $\mathcal{L}^t(\mathcal{D}) = \mathcal{L}$, resp. $\mathcal{L}(\mathcal{D}) \subseteq \mathcal{L}$). From Lemma 1.1 we therefore get

<u>Theorem 1.1</u>. For each multitape AFA $\mathcal{D}$, $\mathcal{L}^t(\mathcal{D})$ and $\mathcal{L}(\mathcal{D})$ are AFL containing $\{\epsilon\}$, with $\mathcal{L}(\mathcal{D})$ being full. Furthermore, for each ν, $\mathcal{L}^t(\mathcal{D}_\nu)$ and $\mathcal{L}(\mathcal{D}_\nu)$ are AFL containing $\{\epsilon\}$, with $\mathcal{L}(\mathcal{D}_\nu)$ being full.

One of the operations upon families of languages in which we shall be interested is intersection. This leads to

<u>Notation</u>. For families of languages $\mathcal{L}_1, \ldots, \mathcal{L}_n$ let

$$\mathcal{L}_1 \wedge \ldots \wedge \mathcal{L}_n = \{L_1 \cap \ldots \cap L_n / \text{each } L_i \text{ in } \mathcal{L}_i\}.$$

We now introduce some notation to describe certain families of languages related to a given family of languages.

<u>Notation</u>. For each family of languages $\mathcal{L}$ let

---

(6) For each set of words A, $A^+ = \bigcup_{i=1}^{\infty} A^i$ and $A^* = \bigcup_{i=0}^{\infty} A^i$, where $A^{i+1} = A^iA$, $i \geq 1$, and $A^0 = \{\epsilon\}$.

(7) A mapping h from Σ_1^* into Σ_2^* is a <u>homomorphism</u> if $h(xy) = h(x)h(y)$ for all x and y in Σ_1^*. If $h(x) = \epsilon$ implies $x=\epsilon$, then h is said to be <u>ϵ-free</u>. The mapping h^{-1} of subsets of Σ_2^* into subsets of Σ_1^* defined by $h^{-1}(Y) = \{x/h(x) \text{ in } Y\}$ for all $Y \subseteq \Sigma_2^*$ is called an <u>inverse</u> homomorphism.

(a) $\mathcal{F}(\mathcal{L})$ be the smallest AFL containing $\mathcal{L}$.

(b) $\hat{\mathcal{F}}(\mathcal{L})$ be the smallest full AFL containing $\mathcal{L}$.

(c) $\mathcal{H}(\mathcal{L}) = \{h(L)/L \text{ in } \mathcal{L}, h \text{ an } \epsilon\text{-free homomorphism}\}$.

(d) $\hat{\mathcal{H}}(\mathcal{L}) = \{h(L)/L \text{ in } \mathcal{L}, h \text{ an arbitrary homomorphism}\}$.

(e) $\mathcal{F}_\cap(\mathcal{L})$ be the smallest AFL containing $\mathcal{L}$ and closed under intersection.

(f) $\hat{\mathcal{F}}_\cap(\mathcal{L})$ be the smallest full AFL containing $\mathcal{L}$ and closed under intersection.

(g) $\wedge \mathcal{L} = \{L_1 \cap \dots \cap L_n/n \geq 1, \text{ each } L_i \text{ in } \mathcal{L}\} = \bigcup_{n\geq 1} (\mathcal{L}_1 \wedge \dots \wedge \mathcal{L}_n)$, $\mathcal{L}_i = \mathcal{L}$ for each i.

Clearly the families in (a), (b), (e), and (f) exist.

Finally, we summarize a number of AFL relations (some already known) which are used extensively in the sequel.

Theorem 1.2. Let $\mathcal{L}, \mathcal{L}_1, \dots, \mathcal{L}_n, \mathcal{L}_1', \dots, \mathcal{L}_m'$ be AFL.

(a) $\hat{\mathcal{H}}(\mathcal{L}) = \hat{\mathcal{F}}(\mathcal{L})$.

(b) $\mathcal{H}(\mathcal{L}_1 \wedge \dots \wedge \mathcal{L}_n) = \mathcal{F}(\mathcal{L}_1 \wedge \dots \wedge \mathcal{L}_n)$, and

$\hat{\mathcal{H}}(\mathcal{L}_1 \wedge \dots \wedge \mathcal{L}_n) = \hat{\mathcal{H}}(\mathcal{H}(\mathcal{L}_1 \wedge \dots \wedge \mathcal{L}_n)) = \hat{\mathcal{F}}(\mathcal{L}_1 \wedge \dots \wedge \mathcal{L}_n)$.

(c) $\mathcal{H}(\mathcal{L}_1 \wedge \dots \wedge \mathcal{L}_n) \wedge \mathcal{H}(\mathcal{L}_1' \wedge \dots \wedge \mathcal{L}_m') \subseteq \mathcal{H}(\mathcal{L}_1 \wedge \dots \wedge \mathcal{L}_n \wedge \mathcal{L}_1' \wedge \dots \wedge \mathcal{L}_m')$

and $\hat{\mathcal{H}}(\mathcal{L}_1 \wedge \dots \wedge \mathcal{L}_n) \wedge \hat{\mathcal{H}}(\mathcal{L}_1' \wedge \dots \wedge \mathcal{L}_m') \subseteq \hat{\mathcal{H}}(\mathcal{L}_1 \wedge \dots \wedge \mathcal{L}_n \wedge \mathcal{L}_1' \wedge \dots \wedge \mathcal{L}_m')$.

(d) $\hat{\mathcal{H}}(\mathcal{L}_1 \wedge \dots \wedge \mathcal{L}_n) = \hat{\mathcal{H}}(\hat{\mathcal{H}}(\mathcal{L}_1) \wedge \dots \wedge \hat{\mathcal{H}}(\mathcal{L}_n))$.

(e) $\mathcal{F}_\cap(\mathcal{L}) = \mathcal{H}(\wedge\mathcal{L}) = \mathcal{F}(\wedge\mathcal{L})$, and

$\hat{\mathcal{F}}_\cap(\mathcal{L}) = \hat{\mathcal{H}}(\wedge\mathcal{L}) = \hat{\mathcal{F}}(\wedge\mathcal{L}) = \hat{\mathcal{F}}(\hat{\mathcal{H}}(\mathcal{L})) = \hat{\mathcal{H}}(\wedge\hat{\mathcal{H}}(\mathcal{L})) = \hat{\mathcal{F}}_\cap(\hat{\mathcal{H}}(\mathcal{L}))$.

Proof. (a) This equality is in [11].

(b) The first equality is in [7] and the second follows from the first.

(c) Let L_1 be in $\mathcal{L}_1 \wedge \ldots \wedge \mathcal{L}_n$ and L_2 in $\mathcal{L}'_1 \wedge \ldots \wedge \mathcal{L}'_m$. Let h_1 and h_2 be homomorphisms, and let $L = h_1(L_1) \cap h_2(L_2)$. Since $f(X) \cap Y = f[X \cap f^{-1}(Y)]$ for arbitrary sets X and Y and an arbitrary function f, $L = h_1(L_1 \cap h_1^{-1} h_2(L_2))$. Since $\mathcal{H}(\mathcal{L}'_1 \wedge \ldots \wedge \mathcal{L}'_m)$ and $\hat{\mathcal{H}}(\mathcal{L}'_1 \wedge \ldots \wedge \mathcal{L}'_m)$ are AFL, $h_1^{-1} h_2(L) = h_3(L'_2)$ for some homomorphism h_3, ϵ-free if h_2 is ϵ-free, and some set L'_2 in $\mathcal{L}'_1 \wedge \ldots \wedge \mathcal{L}'_m$. Then $L = h_1(L_1 \cap h_3(L'_2)) = h_1 h_3(h_3^{-1}(L_1) \cap L'_2)$. From [7], $\mathcal{L}_1 \wedge \ldots \wedge \mathcal{L}_n$ is closed under h^{-1}. Thus L is in $\hat{\mathcal{H}}(\mathcal{L}_1 \wedge \ldots \wedge \mathcal{L}_n \wedge \mathcal{L}'_1 \wedge \ldots \wedge \mathcal{L}'_m)$. If h_1 and h_2 are ϵ-free, then $h_1 h_3$ is ϵ-free so that L is in $\mathcal{H}(\mathcal{L}_1 \wedge \ldots \wedge \mathcal{L}_n \wedge \mathcal{L}'_1 \wedge \ldots \wedge \mathcal{L}'_m)$.

(d) Clearly $\hat{\mathcal{H}}(\mathcal{L}_1) = \hat{\mathcal{H}}[\hat{\mathcal{H}}(\mathcal{L}_1)]$. Continuing by induction, suppose $\hat{\mathcal{H}}(\mathcal{L}_1 \wedge \ldots \wedge \mathcal{L}_{n-1}) = \hat{\mathcal{H}}[\hat{\mathcal{H}}(\mathcal{L}_1) \wedge \ldots \wedge \hat{\mathcal{H}}(\mathcal{L}_{n-1})]$, $n \geq 2$. Consider $\hat{\mathcal{H}}(\mathcal{L}_1 \wedge \ldots \wedge \mathcal{L}_n)$. Obviously $\hat{\mathcal{H}}(\mathcal{L}_1 \wedge \ldots \wedge \mathcal{L}_n) \subseteq \hat{\mathcal{H}}[\hat{\mathcal{H}}(\mathcal{L}_1) \wedge \ldots \wedge \hat{\mathcal{H}}(\mathcal{L}_n)]$. Furthermore,

$$\begin{aligned}
\hat{\mathcal{H}}[\hat{\mathcal{H}}(\mathcal{L}_1) \wedge \ldots \wedge \hat{\mathcal{H}}(\mathcal{L}_n)] &\subseteq \hat{\mathcal{H}}[\hat{\mathcal{H}}(\hat{\mathcal{H}}(\mathcal{L}_1) \wedge \ldots \wedge \hat{\mathcal{H}}(\mathcal{L}_{n-1})) \wedge \hat{\mathcal{H}}(\mathcal{L}_n)] \\
&= \hat{\mathcal{H}}[\hat{\mathcal{H}}(\mathcal{L}_1 \wedge \ldots \wedge \mathcal{L}_{n-1}) \wedge \hat{\mathcal{H}}(\mathcal{L}_n)], \text{ by induction,} \\
&\subseteq \hat{\mathcal{H}}[\hat{\mathcal{H}}(\mathcal{L}_1 \wedge \ldots \wedge \mathcal{L}_{n-1} \wedge \mathcal{L}_n)], \text{ by (c),} \\
&= \hat{\mathcal{H}}(\mathcal{L}_1 \wedge \ldots \wedge \mathcal{L}_n).
\end{aligned}$$

Thus $\hat{\mathcal{H}}(\mathcal{L}_1 \wedge \ldots \wedge \mathcal{L}_n) = \hat{\mathcal{H}}[\hat{\mathcal{H}}(\mathcal{L}_1) \wedge \ldots \wedge \hat{\mathcal{H}}(\mathcal{L}_n)]$.

(e) Consider the first sequence of equalities. Clearly $\mathcal{H}(\wedge\mathcal{L}) = \mathcal{F}(\wedge\mathcal{L}) \subseteq \mathcal{F}_\cap(\mathcal{L})$. To get equality, it suffices to show that $\mathcal{H}(\wedge\mathcal{L})$ is closed under intersection. To this end, note that

$$\begin{aligned}
\mathcal{H}(\wedge\mathcal{L}) \wedge \mathcal{H}(\wedge\mathcal{L}) &= \bigcup_{m \geq 1} \mathcal{H}[\mathcal{L} \wedge \ldots \wedge \mathcal{L}] \wedge \bigcup_{m \geq 1} \mathcal{H}[\mathcal{L} \wedge \ldots \wedge \mathcal{L}] \\
&= \bigcup_{m,n \geq 1} (\mathcal{H}[\mathcal{L} \wedge \ldots \wedge \mathcal{L}]_n \wedge \mathcal{H}[\mathcal{L} \wedge \ldots \wedge \mathcal{L}]_m) \\
&\subseteq \bigcup_{m,n \geq 1} \mathcal{H}[\mathcal{L} \wedge \ldots \wedge \mathcal{L}]_{m+n} \text{ ,by (c),} \\
&\subseteq \mathcal{H}(\wedge\mathcal{L}).
\end{aligned}$$

For the second sequence of equalities, we have $\hat{\mathcal{F}}_{\cap}(\mathcal{L}) = \hat{\mathcal{H}}(\wedge\mathcal{L}) = \hat{\mathcal{F}}(\wedge\mathcal{L})$. Replacing $\mathcal{L}$ by $\hat{\mathcal{H}}(\mathcal{L})$, we have $\hat{\mathcal{F}}_{\cap}(\hat{\mathcal{H}}(\mathcal{L})) = \hat{\mathcal{H}}(\wedge\hat{\mathcal{H}}(\mathcal{L})) = \hat{\mathcal{F}}(\wedge\hat{\mathcal{H}}(\mathcal{L}))$. From (d), we readily get $\hat{\mathcal{H}}(\wedge\mathcal{L}) = \hat{\mathcal{H}}(\wedge\hat{\mathcal{H}}(\mathcal{L}))$. Hence the result.

Corollary. (a) If $n \geq 2$ and $\mathcal{L}_1, \ldots, \mathcal{L}_n$ are AFL, then

$$\mathcal{H}[\mathcal{H}(\mathcal{L}_1 \wedge \ldots \wedge \mathcal{L}_{n-1}) \wedge \mathcal{L}_n] = \mathcal{H}(\mathcal{L}_1 \wedge \ldots \wedge \mathcal{L}_n).$$

(b) If $n \geq 2$ and $\mathcal{L}_1, \ldots, \mathcal{L}_n$ are full AFL, then

$$\hat{\mathcal{H}}[\hat{\mathcal{H}}(\mathcal{L}_1 \wedge \ldots \wedge \mathcal{L}_{n-1}) \wedge \mathcal{L}_n] = \hat{\mathcal{H}}(\mathcal{L}_1 \wedge \ldots \wedge \mathcal{L}_n).$$

Proof. It suffices to show (a), a similar argument holding for (b). Since $\mathcal{L}_1 \wedge \ldots \wedge \mathcal{L}_{n-1} \subseteq \mathcal{H}(\mathcal{L}_1 \wedge \ldots \wedge \mathcal{L}_{n-1})$,

$$\mathcal{H}(\mathcal{L}_1 \wedge \ldots \wedge \mathcal{L}_n) \subseteq \mathcal{H}[\mathcal{H}(\mathcal{L}_1 \wedge \ldots \wedge \mathcal{L}_{n-1}) \wedge \mathcal{L}_n].$$

On the other hand,

$$\begin{aligned}\mathcal{H}[\mathcal{H}(\mathcal{L}_1 \wedge \ldots \wedge \mathcal{L}_{n-1}) \wedge \mathcal{L}_n] &= \mathcal{H}[\mathcal{H}(\mathcal{L}_1 \wedge \ldots \wedge \mathcal{L}_{n-1}) \wedge \mathcal{H}(\mathcal{L}_n)] \\ &\subseteq \mathcal{H}[\mathcal{H}(\mathcal{L}_1 \wedge \ldots \wedge \mathcal{L}_n)], \text{ by (c) of Theorem 1.2,} \\ &= \mathcal{H}(\mathcal{L}_1 \wedge \ldots \wedge \mathcal{L}_n).\end{aligned}$$

Combining, we get the desired equality.

Section 2. Multitape AFA and Intersection

In this section we represent the smallest AFL containing the intersection of languages from a finite sequence of AFL in terms of a multitape AFA. In particular, we define $\mathcal{D}_1 \wedge \ldots \wedge \mathcal{D}_n$ for the sequence of multitape AFA $\mathcal{D}_1, \ldots, \mathcal{D}_n$. (The operator $\wedge$ for multitape AFA plays a basic role throughout the paper and is analogous to the cross product operation $\times$ in set theory.) We then show that $\mathcal{H}(\mathcal{L}^t(\mathcal{D}_1) \wedge \ldots \wedge \mathcal{L}^t(\mathcal{D}_n)) = \mathcal{L}^t(\mathcal{D}_1 \wedge \ldots \wedge \mathcal{D}_n)$ and $\hat{\mathcal{H}}(\mathcal{L}(\mathcal{D}_1) \wedge \ldots \wedge \mathcal{L}(\mathcal{D}_n)) = \mathcal{L}(\mathcal{D}_1 \wedge \ldots \wedge \mathcal{D}_n)$.

We now introduce the operation $\wedge$ for multitape AFA.

<u>Notation</u>. Let $\mathscr{B}$ be an abstract set, $<_{\mathscr{B}}$ a simple order on $\mathscr{B}$, and $\{(\Omega_\beta,\mathscr{D}_\beta)/\beta \text{ in } \mathscr{B}\}$ a family of multitape AFA, with $\Omega_\beta = (K,\Sigma,\mathcal{a}_\beta,<_\beta,\mu_\beta)$ for each β. Then $\wedge_{\mathscr{B}}\mathscr{D}_\beta$ is the multitape AFA $(\Omega,\wedge_{\mathscr{B}}\mathscr{D}_\beta)$, where $\Omega = (K,\Sigma,\mathcal{a},<,\mu)$ is defined as follows:

(1) $\mathcal{a} = \bigcup_{\beta \text{ in } \mathscr{B}} (\mathcal{a}_\beta \times \{\beta\})$.

(2) $\mu(\alpha,\beta) = \mu_\beta(\alpha)$ for each β in $\mathscr{B}$ and each α in $\mathcal{a}_\beta$.

(3) $(\alpha,\beta) < (\alpha',\beta')$ if and only if either $\beta = \beta'$ and $\alpha <_\beta \alpha'$, or $\beta <_{\mathscr{B}} \beta'$.

If $\mathscr{B}$ is a finite set $\mathscr{B} = \{\beta_1, \ldots, \beta_n\}$ and $<_{\mathscr{B}}$ is the order on $\mathscr{B}$ as given, then $\wedge_{\mathscr{B}}\mathscr{D}_{\mathscr{B}}$ is written as $\bigwedge_{1\le i\le n} \mathscr{D}_{\beta_i}$ or $\mathscr{D}_{\beta_1} \wedge \ldots \wedge \mathscr{D}_{\beta_n}$.

Obviously the set of words accepted is independent of the order $<_{\mathscr{B}}$ on $\mathscr{B}$.

Frequently $\mathscr{B}$ is a subset of the integers. In this case, unless stated otherwise, $<_{\mathscr{B}}$ is the natural order of the integers.

If the $\mathcal{a}_\beta$ are pairwise disjoint, then we may identify each (α,β) with α and write $\mathcal{a}$ as $\bigcup_{\beta \text{ in } \mathscr{B}} \mathcal{a}_\beta$. In the sequel, we shall always assume (without loss of generality) that the $\mathcal{a}_\beta$ are pairwise disjoint.

For each multitape AFA $(\Omega,\mathscr{D})$, $\mathscr{D}$ may be identified with $\bigwedge_{\alpha \text{ in } \mathcal{a}} \mathscr{D}_\alpha$.

We now turn toward showing that $\mathcal{L}(\mathscr{D}_1 \wedge \ldots \wedge \mathscr{D}_n) = \hat{\mathcal{H}}(\mathcal{L}(\mathscr{D}_1) \wedge \ldots \wedge \mathcal{L}(\mathscr{D}_n))$ and $\mathcal{L}^t(\mathscr{D}_1 \wedge \ldots \wedge \mathscr{D}_n) = \mathcal{H}(\mathcal{L}^t(\mathscr{D}_1) \wedge \ldots \wedge \mathcal{L}^t(\mathscr{D}_n))$. That is, the family of those sets accepted by at least one acceptor (quasi-realtime acceptor) in the AFA $\mathscr{D}_1 \wedge \ldots \wedge \mathscr{D}_n$ coincides with the family of the homomorphic (ϵ-free homomorphic) images of the sets in $\mathcal{L}(\mathscr{D}_1) \wedge \ldots \wedge \mathcal{L}(\mathscr{D}_n)$ $(\mathcal{L}^t(\mathscr{D}_1) \wedge \ldots \wedge \mathcal{L}^t(\mathscr{D}_n))$. First though, we need two lemmas.

Lemma 2.1. For all multitape AFA $\mathcal{D}_1$ and $\mathcal{D}_2$,

$$\mathcal{L}(\mathcal{D}_1) \wedge \mathcal{L}(\mathcal{D}_2) \subseteq \mathcal{L}(\mathcal{D}_1 \wedge \mathcal{D}_2)$$

and $\mathcal{L}^t(\mathcal{D}_1) \wedge \mathcal{L}^t(\mathcal{D}_2) \subseteq \mathcal{L}^t(\mathcal{D}_1 \wedge \mathcal{D}_2)$.

Proof. For each i, let $\Omega_i = (K,\Sigma,\mathcal{Q}_i,<_i,\mu_i)$ and let $D_i = (K_1,\Sigma_1,\delta_1,q_1,F_1,\nu_1)$ be in $\mathcal{D}_i$, with $\nu_1 = (\alpha_1, \ldots, \alpha_k)$ and $\nu_2 = (\beta_1, \ldots, \beta_\ell)$. Let $\nu_3 = (\alpha_1, \ldots, \alpha_k, \beta_1, \ldots, \beta_\ell)$, $K_3 = K_1 \times K_2$, and $F_3 = F_1 \times F_2$.

Let $D_3 = (K_3,\Sigma_1 \cap \Sigma_2,\delta_3,q_3,F_3,\nu_3)$, where δ_3 is defined as follows:

(1) If a is in $\Sigma_1 \cap \Sigma_2$, $(q',(u_1, \ldots, u_k))$ is in $\delta_1(q,a,(\gamma_1, \ldots, \gamma_k))$ and $(p',(\bar{u}_1, \ldots, \bar{u}_\ell))$ is in $\delta_2(p,a,(\gamma_1', \ldots, \gamma_\ell'))$, let $((q',p'), (u_1, \ldots, u_k, \bar{u}_1, \ldots, \bar{u}_\ell))$ be in $\delta_3((q,p),a,(\gamma_1, \ldots, \gamma_k, \gamma_1', \ldots, \gamma_\ell'))$.

(2) If $(q',(u_1, \ldots, u_k))$ is in $\delta_1(q,\epsilon,(\gamma_1, \ldots, \gamma_k))$, then for all $(\gamma_1', \ldots, \gamma_\ell')$ in G_{D_2} and p in K_2 let $((q',p), (u_1, \ldots, u_k, 1(\beta_1,\gamma_1'), \ldots, 1(\beta_\ell,\gamma_\ell')))$ be in $\delta_3((q,p),\epsilon,(\gamma_1, \ldots, \gamma_k, \gamma_1', \ldots, \gamma_\ell'))$.

(3) If $(p',(u_1', \ldots, u_\ell'))$ is in $\delta_2(p,\epsilon,(\gamma_1', \ldots, \gamma_\ell'))$, then for all $(\gamma_1, \ldots, \gamma_k)$ in G_{D_1} and all q in K_1 let $((q,p'),(1(\alpha_1,\gamma_1), \ldots, 1(\alpha_k,\gamma_k),u_1', \ldots, u_\ell'))$ be in $\delta_3((q,p),\epsilon,(\gamma_1, \ldots, \gamma_k, \gamma_1', \ldots, \gamma_\ell'))$.

Clearly $L(D_3) = L(D_1) \cap L(D_2)$ and D_3 is quasi-realtime if and only if D_1 and D_2 are quasi-realtime. Hence the result.

Lemma 2.2. For all multitape AFA $\mathcal{D}_1$ and $\mathcal{D}_2$

(a) $\mathcal{H}(\mathcal{L}^t(\mathcal{D}_1) \wedge \mathcal{L}^t(\mathcal{D}_2)) = \mathcal{L}^t(\mathcal{D}_1 \wedge \mathcal{D}_2)$

and (b) $\hat{\mathcal{H}}(\mathcal{L}(\mathcal{D}_1) \wedge \mathcal{L}(\mathcal{D}_2)) = \hat{\mathcal{H}}(\mathcal{L}^t(\mathcal{D}_1) \wedge \mathcal{L}^t(\mathcal{D}_2)) = \mathcal{L}(\mathcal{D}_1 \wedge \mathcal{D}_2)$.

Proof. For each i, let $\Omega_i = (K,\Sigma,\mathcal{Q}_i,<_i,\mu_i)$. Consider (a). By Lemma 2.1 and Theorem 1.1, $\mathcal{H}(\mathcal{L}^t(\mathcal{D}_1) \wedge \mathcal{L}^t(\mathcal{D}_2)) \subseteq \mathcal{H}[\mathcal{L}^t(\mathcal{D}_1 \wedge \mathcal{D}_2)] = \mathcal{L}^t(\mathcal{D}_1 \wedge \mathcal{D}_2)$.

To see the reverse containment let $D = (K_1,\Sigma_1,\delta,q_o,F,\nu)$ be quasi-realtime and in

$\mathfrak{P}_3$, with $\upsilon = (\alpha_1, \ldots, \alpha_k, \beta_1, \ldots, \beta_\ell)$, $\alpha_1, \ldots, \alpha_k$ in $\mathcal{a}_1$, and $\beta_1, \ldots, \beta_\ell$ in $\mathcal{a}_2$. Let $L = L(D)$. For each $q, a, \gamma_1, \ldots, \gamma_{k+\ell}$ such that $\#(\delta(q,a,(\gamma_1, \ldots, \gamma_{k+\ell}))) > 0$[8] and each i, $1 \le i \le \#(\delta(q,a,(\gamma_1, \ldots, \gamma_{k+\ell})))$ let $(q,a,(\gamma_1, \ldots, \gamma_{k+\ell}),i)$ be a new element of Σ. Let Σ_2 be the set of all such $(q,a,(\gamma_1, \ldots, \gamma_{k+\ell}),i)$. Clearly Σ_2 is finite. Let c be a new element of Σ. Let h_1 and h_2 be the homomorphisms on Σ_2^* and $(\Sigma_1 \cup \{c\})^*$ resp. defined by $h_1((q,a,(\gamma_1, \ldots, \gamma_{k+\ell}),i)) = a$ if $a \ne \epsilon$ and is c if $a = \epsilon$, and $h_2(a) = a$ if $a \ne c$ and $h_2(c) = \epsilon$. We shall show that there exist L_1 in $\mathcal{L}^t(\mathfrak{P}_1)$ and L_2 in $\mathcal{L}^t(\mathfrak{P}_2)$ such that $L = h_2h_1(L_1 \cap L_2)$ and L_2 is ϵ-limited on $h_1(L_1 \cap L_2)$.[9] From this it will follow that $h_1(L_1 \cap L_2)$ is in $\mathcal{H}(\mathcal{L}^t(\mathfrak{P}_1) \wedge \mathcal{L}^t(\mathfrak{P}_2))$, an AFL containing $\{\epsilon\}$ [7], thus that $h_2h_1(L_1 \cap L_2))$ is in $\mathcal{H}(\mathcal{L}^t(\mathfrak{P}_1) \wedge \mathcal{L}^t(\mathfrak{P}_2))$ [5].

For $i = 1,2$ let $D_i = (K_1, \Sigma_2, \delta_i, q_0, F, \upsilon_i)$, where $\upsilon_1 = (\alpha_1, \ldots, \alpha_k)$, $\upsilon_2 = (\beta_1, \ldots, \beta_\ell)$, and δ_i is defined as follows: For each $(q,a,(\gamma_1, \ldots, \gamma_{k+\ell}),i)$ in Σ_2, let the elements of $\delta(q,a,(\gamma_1, \ldots, \gamma_{k+\ell}))$ be simply ordered in some way. If $(q',(u_1, \ldots u_{k+\ell}))$ is the i-th member of $\delta(q,a,(\gamma_1, \ldots, \gamma_{k+\ell}))$, let

$$\delta_1(q,(q,a,(\gamma_1, \ldots, \gamma_{k+\ell}),i), (\gamma_1, \ldots, \gamma_k)) = \{(q',(u_1, \ldots, u_k))\}$$

and $\delta_2(q,(q,a,(\gamma_1, \ldots, \gamma_{k+\ell}),i), (\gamma_{k+1}, \ldots \gamma_{k+\ell})) = \{(q',(u_{k+1}, \ldots, u_{k+\ell}))\}$.

Let $L_1 = L(D_1)$ and $L_2 = L(D_2)$. Since δ_1 and δ_2 have no ϵ-moves, L_1 is in $\mathcal{L}^t(\mathfrak{P}_1)$ and L_2 is in $\mathcal{L}^t(\mathfrak{P}_2)$. Clearly $L \subseteq h_2h_1(L_1 \cap L_2)$.

---

(8) For each set E, $\#(E)$ denotes the number of elements in it.

(9) A homomorphism h is <u>ϵ-limited</u> on a set L if there exists $k \ge 0$ such that for all w in L, if $w = xyz$ and $h(y) = \epsilon$, then $|y| < k$.

Consider the reverse containment. If ϵ is in $L_1 \cap L_2$, then q_0 is in F and ϵ is in L. Suppose $w \neq \epsilon$ is in $L_1 \cap L_2$, with $|w| = n$. Then

$$w = (p_1, a_1, (\gamma_{11}, \ldots, \gamma_{1(k+\ell)}), j_1) \ldots (p_n, a_n, (\gamma_{n1}, \ldots, \gamma_{n(k+\ell)}), j_n)$$

for some $p_1, \ldots, p_n$ in K_1, $a_1, \ldots, a_n$ in $\Sigma_1 \cup \{\epsilon\}$, each $(\gamma_{i1}, \ldots, \gamma_{i(k+\ell)})$ in G_D, and $1 \le j_i \le \#(\delta(p_i, a_i, (\gamma_{i1}, \ldots, \gamma_{i(k+\ell)})))$, $1 \le i \le n$. For each i, let $(p_i', (u_{i1}, \ldots, u_{i(k+\ell)}))$ be the j_i-th element of $\delta(p_i, a_i, (\gamma_{i1}, \ldots, \gamma_{i(k+\ell)}))$. By definition of $\vdash$, δ_1, and δ_2, it follows that $q_0 = p_1, p_r = p'_{r+1}$, $1 \le r < n$, and p_n' is in F. Furthermore, there exists $y_{11}, \ldots, y_{1(k+\ell)}, y_{21}, \ldots, y_{2(k+\ell)}, \ldots, y_{n(k+\ell)}$ such that $y_{11} = \ldots = y_{1(k+\ell)} = \epsilon, y_{(i+1)r} = f_{\alpha_r}(y_{ir}, u_{ir})$ and $\epsilon = f_{\alpha_r}(y_{nr}, u_{nr})$ for $1 \le i < n$, $1 \le r \le k$, and $y_{(i+1)(k+r)} = f_{\beta_r}(y_{i(k+r)}, u_{i(k+r)})$ and $\epsilon = f_{\beta_r}(y_{n(k+r)}, u_{n(k+r)})$ for $1 \le i < n$, $1 \le r \le \ell$. Let $p_{n+1} = p_n'$ and $y_{(n+1)1} = \ldots y_{(n+1)(k+\ell)} = \epsilon$. Then

$$(p_i, a_i, (y_{i1}, \ldots, y_{i(k+\ell)})) \vdash_{\overline{D}} (p_{i+1}, \epsilon, (y_{(i+1)1}, \ldots, y_{(i+1)(k+\ell)}))$$

for $1 \le i \le n$, so that $a_1 \ldots a_n = h_2h_1(w)$ is in L. Therefore $h_2h_1(L_1 \cap L_2) = L$. Furthermore, if

$$h_2h_1((p_r, a_r, (\gamma_{r1}, \ldots, \gamma_{r(k+\ell)}), j_r) \ldots (p_{r+s}, a_{r+s}, (\gamma_{(r+s)1}, \ldots, \gamma_{(r+s)(k+\ell)}),$$

then $h_1((p_r, \ldots) \ldots (p_{r+s}, \ldots)) = c^{s+1}$ and $a_r = \ldots = a_{r+s} = \epsilon$. Then

$$(p_r, \epsilon, (y_{r1}, \ldots, y_{r(k+\ell)})) \vdash_{\overline{D}}^{s+1} (p_{r+s+1}, \epsilon, (y_{(r+s+1)1}, \ldots, y_{(r+s+1)(k+\ell)})).$$

Since D is quasi-realtime, there exists an integer t such that for all configurations $C = (q, \epsilon, (\gamma_1, \ldots, \gamma_{k+\ell}))$ and $C' = (q', \epsilon, (\gamma_1', \ldots, \gamma_{k+\ell}'))$ of D, $C \vdash^{i} C'$ implies $i \le t$. Hence $s+1 \le t$. Therefore h_2 is ϵ-limited on $h_1(L_1 \cap L_2)$.

Consider (b). It was shown in [5] that for any single-tape AFA $\mathfrak{D}$, thus for any multitape AFA $\mathfrak{D}$ by the corollary to Lemma 1.1, $\mathfrak{L}(\mathfrak{D}) = \hat{\mathcal{H}}(\mathfrak{L}^t(\mathfrak{D}))$. Hence

$$\mathcal{L}(\mathfrak{A}_1 \wedge \mathfrak{A}_2) = \hat{\mathcal{H}}(\mathcal{L}^t(\mathfrak{A}_1 \wedge \mathfrak{A}_2))$$
$$= \hat{\mathcal{H}}[\mathcal{H}(\mathcal{L}^t(\mathfrak{A}_1) \wedge \mathcal{L}^t(\mathfrak{A}_2))], \text{ by (a)}$$
$$= \hat{\mathcal{H}}(\mathcal{L}^t(\mathfrak{A}_1) \wedge \mathcal{L}^t(\mathfrak{A}_2))$$
$$= \hat{\mathcal{H}}(\hat{\mathcal{H}}(\mathcal{L}^t(\mathfrak{A}_1)) \wedge \hat{\mathcal{H}}(\mathcal{L}^t(\mathfrak{A}_2))), \text{ by Theorem 1.2 (d),}$$
$$= \hat{\mathcal{H}}(\mathcal{L}(\mathfrak{A}_1) \wedge \mathcal{L}(\mathfrak{A}_2)).$$

Theorem 2.1. For all multitape AFA $\mathfrak{A}_1, \ldots, \mathfrak{A}_n$

(a) $\mathcal{H}(\mathcal{L}^t(\mathfrak{A}_1) \wedge \ldots \wedge \mathcal{L}^t(\mathfrak{A}_n)) = \mathcal{L}^t(\mathfrak{A}_1 \wedge \ldots \wedge \mathfrak{A}_n)$

and (b) $\hat{\mathcal{H}}(\mathcal{L}(\mathfrak{A}_1) \wedge \ldots \wedge \mathcal{L}(\mathfrak{A}_n)) = \hat{\mathcal{H}}(\mathcal{L}^t(\mathfrak{A}_1) \wedge \ldots \wedge \mathcal{L}^t(\mathfrak{A}_n)) = \mathcal{L}(\mathfrak{A}_1 \wedge \ldots \wedge \mathfrak{A}_n)$.

Proof. For each i let $\Omega_i = (K,\Sigma,\mathfrak{a}_i,<_i,\mu_i)$.

(a) Clearly $\mathcal{H}(\mathcal{L}^t(\mathfrak{A}_1)) = \mathcal{L}^t(\mathfrak{A}_1)$, so that (a) is true for $n=1$. Continuing by induction suppose the theorem is true for n-1. Now

$$\mathcal{L}^t(\mathfrak{A}_1 \wedge \ldots \wedge \mathfrak{A}_n) = \mathcal{L}^t((\mathfrak{A}_1 \wedge \ldots \wedge \mathfrak{A}_{n-1}) \wedge \mathfrak{A}_n)$$
$$= \mathcal{H}(\mathcal{L}^t(\mathfrak{A}_1 \wedge \ldots \wedge \mathfrak{A}_{n-1}) \wedge \mathcal{L}^t(\mathfrak{A}_n)), \text{ by Lemma 2.2,}$$
$$= \mathcal{H}[\mathcal{H}(\mathcal{L}^t(\mathfrak{A}_1) \wedge \ldots \wedge \mathcal{L}^t(\mathfrak{A}_{n-1})) \wedge \mathcal{H}(\mathcal{L}^t(\mathfrak{A}_n))], \text{ by induction.}$$

Now $\mathcal{H}[\mathcal{H}(\mathcal{L}^t(\mathfrak{A}_1) \wedge \ldots \wedge \mathcal{L}^t(\mathfrak{A}_{n-1})) \wedge \mathcal{H}(\mathcal{L}^t(\mathfrak{A}_n))]$

$$\subseteq \mathcal{H}[\mathcal{H}(\mathcal{L}^t(\mathfrak{A}_1) \wedge \ldots \wedge \mathcal{L}^t(\mathfrak{A}_{n-1}) \wedge \mathcal{L}^t(\mathfrak{A}_n))], \text{ by Theorem 1.2 c,}$$
$$= \mathcal{H}(\mathcal{L}^t(\mathfrak{A}_1) \wedge \ldots \wedge \mathcal{L}^t(\mathfrak{A}_n))$$
$$\subseteq \mathcal{H}[\mathcal{H}(\mathcal{L}^t(\mathfrak{A}_1) \wedge \ldots \wedge \mathcal{L}^t(\mathfrak{A}_{n-1})) \wedge \mathcal{H}(\mathcal{L}^t(\mathfrak{A}_n))],$$
$$\text{since } \mathcal{L}^t(\mathfrak{A}_1) \wedge \ldots \wedge \mathcal{L}^t(\mathfrak{A}_{n-1}) \subseteq \mathcal{H}(\mathcal{L}^t(\mathfrak{A}_1) \wedge \ldots \wedge \mathcal{L}^t(\mathfrak{A}_{n-1})).$$

Thus we have equality. Hence

$$\mathcal{H}(\mathcal{L}^t(\mathfrak{A}_1) \wedge \ldots \wedge \mathcal{L}^t(\mathfrak{A}_n)) = \mathcal{H}[\mathcal{H}(\mathcal{L}^t(\mathfrak{A}_1) \wedge \ldots \wedge \mathcal{L}^t(\mathfrak{A}_{n-1})) \wedge \mathcal{L}^t(\mathfrak{A}_n))]$$
$$= \mathcal{L}^t(\mathfrak{A}_1 \wedge \ldots \wedge \mathfrak{A}_n).$$

(b) $$\hat{\mathcal{H}}[\mathcal{L}(\mathcal{D}_1) \wedge \ldots \wedge \mathcal{L}(\mathcal{D}_n)] = \hat{\mathcal{H}}[\hat{\mathcal{H}}(\mathcal{L}^t(\mathcal{D}_1)) \wedge \ldots \wedge \hat{\mathcal{H}}(\mathcal{L}^t(\mathcal{D}_n))]$$
$$= \hat{\mathcal{H}}[\mathcal{L}^t(\mathcal{D}_1) \wedge \ldots \wedge \mathcal{L}^t(\mathcal{D}_n)], \text{ by Theorem 1.2 d,}$$
$$= \hat{\mathcal{H}}[\mathcal{H}(\mathcal{L}^t(\mathcal{D}_1) \wedge \ldots \wedge \mathcal{L}^t(\mathcal{D}_n))]$$
$$= \hat{\mathcal{H}}(\mathcal{L}^t(\mathcal{D}_1 \wedge \ldots \wedge \mathcal{D}_n)), \text{ by (a)}$$
$$= \mathcal{L}(\mathcal{D}_1) \wedge \ldots \wedge \mathcal{D}_n).$$

Examples. (1) Let $\mathcal{D}_c$ denote the AFA of 1-counters. Now it is known that every recursively enumerable (r.e) set is accepted by at least one 2-counter [3, 19]. Thus the family of r.e. sets is $\mathcal{L}(\mathcal{D}_c \wedge \mathcal{D}_c)$, which is $\hat{\mathcal{H}}(\mathcal{L}^t(\mathcal{D}_c) \wedge \mathcal{L}^t(\mathcal{D}_c))$ by Theorem 2.1. Examining the proof of Lemma 2.2, we see that every r.e. set is expressible as the homomorphic image of a pair of deterministic realtime 1-counter languages. Since it is undecidable if an arbitrary r.e. set is empty, it is undecidable if $L_1 \cap L_2 = \emptyset$ for arbitrary deterministic realtime 1-counter languages.

(2) Let $\mathcal{D}_p$ be the AFA of pushdown acceptors (pda). The family of list languages defined in [8] is the family of ϵ-free languages[10] in $\mathcal{L}^t(\mathcal{D}_p \wedge \mathcal{D}_p)$. By Theorem 2.1, $\mathcal{L}^t(\mathcal{D}_p \wedge \mathcal{D}_p) = \mathcal{H}(\mathcal{L}^t(\mathcal{D}_p) \wedge \mathcal{L}^t(\mathcal{D}_p))$. Let $\mathcal{L}_1$ be the ϵ-free languages in $\mathcal{L}^t(\mathcal{D}_p)$. Clearly the family of list languages is then $\mathcal{H}(\mathcal{L}_1 \wedge \mathcal{L}_1)$. Let $\mathcal{L}_{\epsilon CF}$ be the family of ϵ-free context-free languages. It is shown in [14] that $\mathcal{L}_{\epsilon CF} = \mathcal{L}_1$. Therefore the family of list languages can be characterized as $\mathcal{H}(\mathcal{L}_{\epsilon CF} \wedge \mathcal{L}_{\epsilon CF}$.) It is proved in [11] that $\mathcal{H}(\mathcal{L}_{\epsilon CF} \wedge \mathcal{L}_{\epsilon CF})$ is the family of languages defined by a context-free "control" set acting on an "ϵ-free" context-free grammar, thereby providing a second characterization of the list languages. A third characterization will appear in Section 3.

---

(10) A set is <u>ϵ-free</u> if it does not contain ϵ.

We now turn to $\mathcal{H}(\wedge\mathcal{L}^t(\mathcal{D}))$ and $\hat{\mathcal{H}}(\wedge\mathcal{L}(\mathcal{D}))$.

<u>Notation</u>. If $\mathcal{D}$ is a multitape AFA, then $\wedge\mathcal{D}$ is the multitape AFA $\wedge_{\mathcal{a}}\mathcal{D}_n$, where $\mathcal{a} = \{n/n\geq 1\}$ and $\mathcal{D}_n = \mathcal{D}$ for each n.

<u>Theorem 2.2</u>. For each single-tape AFA $(\Omega,\mathcal{D})$

(1) $\mathcal{L}^t(\wedge\mathcal{D}) = \mathcal{H}(\wedge\mathcal{L}^t(\mathcal{D})) = \mathcal{F}_\cap(\mathcal{L}^t(\mathcal{D}))$.

(2) $\mathcal{L}(\wedge\mathcal{D}) = \hat{\mathcal{H}}(\wedge\mathcal{L}(\mathcal{D})) = \hat{\mathcal{H}}(\wedge\mathcal{L}^t(\mathcal{D})) = \hat{\mathcal{F}}_\cap(\mathcal{L}(\mathcal{D})) = \hat{\mathcal{F}}_\cap(\mathcal{L}^t(\mathcal{D}))$.

<u>Proof</u>. (1) By Theorem 1.2 e, $\mathcal{H}(\wedge\mathcal{L}^t(\mathcal{D})) = \mathcal{F}_\cap(\mathcal{L}^t(\mathcal{D}))$. Suppose $\Omega = (K,\Sigma,\Gamma,I,f,g)$. Then $\wedge\mathcal{D}$ is the multitape AFA $(\bar{\Omega},\bar{\mathcal{D}})$, where $\bar{\Omega} = (K,\Sigma,\mathcal{a},<,\mu)$, $\mathcal{a} = \{i/i\geq 1\}$, $< \; = \; <$, and $\mu(i) = (\Gamma,I,f,g)$ for each i. Let $\mathcal{D}_n = \bar{\mathcal{D}}_{(1,\ \ldots,\ n)}$. Now for each n, $\mathcal{D}_n$ may be regarded as the n-tape AFA, $\mathcal{D}\wedge\ldots\wedge\mathcal{D}$ (n times). By Theorem 2.1, therefore, $\mathcal{L}^t(\mathcal{D}_n) = \mathcal{H}(\mathcal{L}^t(\mathcal{D})\wedge\ldots\wedge\mathcal{L}^t(\mathcal{D}))$. Thus $\mathcal{L}^t(\mathcal{D}_n) \subseteq \mathcal{H}(\wedge\mathcal{L}^t(\mathcal{D}))$. Then $\mathcal{L}^t(\wedge\mathcal{D}) = \bigcup_n \mathcal{L}^t(\mathcal{D}_n) \subseteq \mathcal{H}(\wedge\mathcal{L}^t(\mathcal{D}))$. On the other hand, if L is in $\mathcal{H}(\wedge\mathcal{L}^t(\mathcal{D}))$, then $L = h(L_1\cap\ldots\cap L_n)$ for some ϵ-free homomorphism h, some $n\geq 1$, and some languages $L_1, \ldots, L_n$ in $\mathcal{L}^t(\mathcal{D})$. Then $L_1\cap\ldots\cap L_n$, thus L, is in $\mathcal{L}^t(\mathcal{D}_n)$. Therefore $\mathcal{H}(\wedge\mathcal{L}^t(\mathcal{D})) \subseteq \mathcal{L}^t(\wedge\mathcal{D})$, whence equality.

(2) Now $\mathcal{L}(\wedge\mathcal{D}) = \hat{\mathcal{H}}(\mathcal{L}(\wedge\mathcal{D})) = \hat{\mathcal{H}}(\mathcal{L}^t(\wedge\mathcal{D}))$

$= \hat{\mathcal{H}}(\mathcal{H}(\wedge\mathcal{L}^t(\mathcal{D})))$, by (1) above

$= \hat{\mathcal{H}}(\wedge\mathcal{L}^t(\mathcal{D}))$

$= \hat{\mathcal{F}}_\cap(\mathcal{L}^t(\mathcal{D}))$, by Theorem 1.2 e.

Then $\mathcal{L}(\wedge\mathcal{D}) = \hat{\mathcal{H}}(\wedge\mathcal{L}^t(\mathcal{D}))$

$= \hat{\mathcal{H}}(\wedge\hat{\mathcal{H}}(\mathcal{L}^t(\mathcal{D})))$, by Theorem 1.2 e,

$= \hat{\mathcal{H}}(\wedge\mathcal{L}(\mathcal{D}))$

$= \hat{\mathcal{F}}_\cap(\mathcal{L}(\mathcal{D}))$, by Theorem 1.2 e.

Hence (2) follows.

Using the previous result, we now present a characterization of a (full) AFL closed under intersection.

Theorem 2.3. $\mathcal{L}$ is a (full) AFL containing $\{\epsilon\}$ and closed under intersection if and only if there exists an AFA $(\Omega,\mathcal{D})$, $\Omega = (K,\Sigma,\mathcal{A},<,\mu)$, such that $\mathcal{L} = \mathcal{L}^t(\mathcal{D})$ $(\mathcal{L} = \mathcal{L}(\mathcal{D}))$, $\mathcal{A}$ is infinite, and $\mu(\alpha) = \mu(\beta)$ for all α and β in $\mathcal{A}$.

Proof. Suppose $(\Omega,\mathcal{D})$ is an AFA such that $\mathcal{L} = \mathcal{L}^t(\mathcal{D})$ $(\mathcal{L} = \mathcal{L}(\mathcal{D}))$, $\mathcal{A}$ is infinite, and $\mu(\alpha) = \mu(\beta)$ for all α and β in $\mathcal{A}$. Clearly $\mathcal{L}$ contains $\{\epsilon\}$. From the $\mathcal{L}^t(\mathcal{D})$ and $\mathcal{L}(\mathcal{D})$ point of view, there is no loss in assuming $\mathcal{A}$ is countable. Then $\mathcal{D} = \bigwedge_{\mathcal{A}} \mathcal{D}_\alpha = \wedge \mathcal{D}_1$, where $\mathcal{D}_\alpha = \mathcal{D}_1$ for all α. The "if" then follows from Theorem 2.2.

Consider the "only if." Suppose $\mathcal{L}$ is a (full) AFL containing $\{\epsilon\}$ and closed under intersection. Hence there exists a single-tape AFA $\mathcal{D}_1$ such that $\mathcal{L}^t(\mathcal{D}_1) = \mathcal{L}$ $(\mathcal{L}(\mathcal{D}_1) = \mathcal{L})$. By Theorem 2.2, $\mathcal{L}^t(\wedge \mathcal{D}_1) = \mathcal{F}_\cap(\mathcal{L}^t(\mathcal{D}_1)) = \mathcal{F}_\cap(\mathcal{L}) = \mathcal{L}$ since $\mathcal{L}$ is an AFL closed under intersection $(\mathcal{L}(\wedge \mathcal{D}_1) = \hat{\mathcal{F}}_\cap(\mathcal{L}(\mathcal{D}_1)) = \hat{\mathcal{F}}_\cap(\mathcal{L}) = \mathcal{L})$. The result then follows from the fact that $\wedge \mathcal{D}_1$ is an AFA satisfying the theorem.

Examples. (1) Let $\mathcal{D}_p$ be the family of pda and $\mathcal{D}_T$ the family of single-tape one-way Turing acceptors (i.e., the input tape is read one way). It is known that each Turing acceptor can be imitated, without loss of time, by some 2-pushdown acceptor [3]. Thus $\mathcal{L}^t(\mathcal{D}_p) \subseteq \mathcal{L}^t(\mathcal{D}_T) \subseteq \mathcal{L}^t(\mathcal{D}_p \wedge \mathcal{D}_p) = \mathcal{H}(\mathcal{L}^t(\mathcal{D}_p) \wedge \mathcal{L}^t(\mathcal{D}_p))$. Then $\mathcal{H}(\wedge \mathcal{L}^t(\mathcal{D}_p)) \subseteq \mathcal{H}(\wedge \mathcal{L}^t(\mathcal{D}_T)) \subseteq \mathcal{H}(\wedge \mathcal{L}^t(\mathcal{D}_p))$, whence equality. Therefore

$$\mathcal{L}^t(\wedge \mathcal{D}_p) = \mathcal{H}(\wedge \mathcal{L}^t(\mathcal{D}_p)) = \mathcal{H}(\wedge \mathcal{L}^t(\mathcal{D}_T)) = \mathcal{L}^t(\wedge \mathcal{D}_T).$$

In other words, L can be recognized in quasi-realtime by a multitape Turing acceptor if and only if L can be recognized in quasi-realtime by a multi-pushdown tape acceptor if and only if L is the ϵ-free homomorphic image of the (finite) intersection of context-free languages.

(2) By a straightforward extension of results in [15] and [4], it can be shown that for each $n\geq 2$,

$$L_n = \{a^{m_1}b \ldots a^{m_n}c\, a^{m_n}b \ldots a^{m_1}/m_1, \ldots, m_n \geq 1\}$$

is recognized by a quasi-realtime n-counter acceptor but by no (n-1)-counter acceptor. That is, if $\mathcal{D}_c$ is the AFA of 1-counter acceptors and $\mathcal{D}_i = \mathcal{D}_c$ for each $i\geq 1$, then L_n is in $\mathcal{L}^t(\mathcal{D}_1 \wedge \ldots \wedge \mathcal{D}_n)$ but not in $\mathcal{L}^t(\mathcal{D}_1 \wedge \ldots \wedge \mathcal{D}_{n-1})$. Thus

$$\begin{aligned} \mathcal{L}^t(\mathcal{D}_c) \subset \mathcal{L}^t(\mathcal{D}_c \wedge \mathcal{D}_c) &= \mathcal{H}(\mathcal{L}^t(\mathcal{D}_c) \wedge \mathcal{L}^t(\mathcal{D}_c)) \\ &\subset \mathcal{H}(\mathcal{L}^t(\mathcal{D}_c) \wedge \mathcal{L}^t(\mathcal{D}_c) \wedge \mathcal{L}^t(\mathcal{D}_c)) \\ &\ldots, \end{aligned}$$

with each containment proper, forms an infinite hierarchy of AFL properly contained in $\mathcal{L}_{CS}$, the family of context-sensitive languages. By contrast it is still open whether the family of list languages, the ϵ-free languages in $\mathcal{L}^t(\mathcal{D}_p \wedge \mathcal{D}_p) = \mathcal{H}(\mathcal{L}^t(\mathcal{D}_p) \wedge \mathcal{L}^t(\mathcal{D}_p))$, is properly contained in the family of ϵ-free languages of $\mathcal{H}(\mathcal{L}^t(\mathcal{D}_p) \wedge \mathcal{L}^t(\mathcal{D}_p) \wedge \mathcal{L}^t(\mathcal{D}_p))$ or whether the family of ϵ-free languages in $\mathcal{L}^t(\mathcal{D}_T)$ is <u>properly</u> contained in $\mathcal{L}_{CS}$.

In passing, we note below a specialized result between the $\wedge$ operation, linear homomorphisms, and AFL.

<u>Definition</u>. A homomorphism h is <u>linear</u> on $L \subseteq \Sigma_1^*$ if there exists $k>0$ such that $|w| \leq k|h(w)|$ for all w in L. For each family of languages $\mathcal{L}$, let

$$\mathcal{H}^{lin}(\mathcal{L}) = \{h(L)/L \text{ in } \mathcal{L},\ h \text{ linear on } \mathcal{L}\}.$$

It is shown in [9] that $\mathcal{H}^{lin}(\mathcal{L})$ is an AFL for each AFL $\mathcal{L}$.

Theorem 2.4. Given d_1, d_2, and d_3, $\mathcal{H}^{lin}(\wedge\,\mathcal{L}) = \mathcal{H}^{lin}(\mathcal{L} \wedge \mathcal{F}(L) \wedge \mathcal{F}(L))$(11) for every ϵ-free AFL $\mathcal{L}$(12) containing $L = \{wd_3w^R/w \text{ in } \{d_1,d_2\}^*\}$(13).

Proof. Since L is in $\mathcal{L}$, $\mathcal{H}^{lin}(\mathcal{L}\wedge\mathcal{F}(L) \wedge\mathcal{F}(L)) \subseteq \mathcal{H}^{lin}(\wedge\mathcal{L})$. It thus suffices to show the reverse inclusion. Therefore let $L_1, \ldots, L_n$ be in $\mathcal{L}$, with $\bigcup_1^n L_i \subseteq \Sigma_1^*$, Σ_1 finite. Let c be a symbol not in Σ_1. Since $\mathcal{F}(L)$ contains $L = \{wd_3w^R/w \text{ in } \{d_1,d_2\}^*\}$, $\mathcal{F}(L)$ contains the ϵ-free linear context-free languages. Therefore $\mathcal{F}(L)$ contains the linear context-free language $L_{\Sigma_1} = \{wcw^R/w \text{ in } \Sigma_1^*\}$. Let $S_1 = (L_{\Sigma_1}c)^n$, $S_2 = \Sigma_1^*c(L_{\Sigma_1}c)^{n-1}\Sigma_1^*c$, and $L' = (L_1\, c\Sigma_1^*c) \ldots (L_n c\Sigma_1^*c)$. Then $S_1 \cap S_2 = \{(wcw^Rc)^n/w \text{ in } \Sigma_1^*\}$,

$$L' \cap S_1 \cap S_2 = \{(wcw^Rc)^n/w \text{ in } L_1 \cap \ldots \cap L_n\},$$

and $S_1 \cap S_2$ is in $\mathcal{F}(L)\wedge\mathcal{F}(L)$. For each a in Σ_1, let $\bar{a}$ be a new symbol and $\Sigma_2 = \{\bar{a}/a \text{ in } \Sigma_1\}$. Let h_1, h_2, and h_3 be the homomorphisms defined by $h_1(a) = h_1(\bar{a}) = a$, $h_1(c) = c$, $h_2(a) = a$, $h_2(\bar{a}) = h_2(c) = \epsilon$, and $h_3(a) = \bar{a}$, for all a in Σ_1. Then

$$L'' = h_1^{-1}(L') \cap h_1^{-1}(S_1) \cap \Sigma_1^*c(\Sigma_2 \cup \{c\})^*$$

$$= \{wch_3(w^R)c\,[h_3(w)ch_3(w^R)c]^{n-1}/w \text{ in } L_1 \cap \ldots \cap L_n\}.$$

Since $\mathcal{L}$ is ϵ-free, h_2 is linear on L'' and $L_1 \cap \ldots \cap L_n = h_2(L'')$ is in $\mathcal{H}^{lin}(\mathcal{L}\wedge\mathcal{F}(L)\wedge\mathcal{F}(L))$. Hence $\mathcal{H}^{lin}(\wedge\mathcal{L}) \subseteq \mathcal{H}^{lin}(\mathcal{L}\wedge\mathcal{F}(L)\wedge\mathcal{F}(L))$, whence equality.

---

(11) We write $\mathcal{F}(L)$ for $\mathcal{F}(\{L\})$.

(12) A family of languages $\mathcal{L}$ is <u>ϵ-free</u> if each language in $\mathcal{L}$ is ϵ-free.

(13) Let $\epsilon^R = \epsilon$ and $(a_1 \ldots a_n)^R = a_n \ldots a_1$, each a_i a symbol.

Section 3. Multitape Transducers

In the previous section we established some connections between multitape AFA, AFL, and the $\wedge$ operations for AFL and multitape AFA. In this section we add an output tape to a multitape AFA to obtain an associated family of multitape transducers. We then note connections between multitape transducers, composition of single-tape transducers, and the $\wedge$ operation for multitape AFA.

We first define multitape transducers.

Definition. Let $(\Omega,\mathcal{D})$ be a multitape AFA, with $\Omega = (K,\Sigma,\mathcal{a},<,\mu)$. Let $(\Omega,\mathcal{D}^o)$ or $\mathcal{D}^o$ when Ω is understood, be the set of all 6-tuples $M = (K_1,\Sigma_1,\Sigma_2,\delta,q_o,\upsilon)$, called multitape transducers, such that

(a) K_1, Σ_1, and Σ_2 are finite nonempty subsets of K, Σ, and Σ, resp.

(b) q_o is in K_1.

(c) $\upsilon = (\alpha_1, \ldots, \alpha_k)$, k finite, α_i in $\mathcal{a}$ for each i, and $\alpha_i < \alpha_{i+1}$ for $1 \leq i < k$.

(d) δ is a function from $K_1 \times \Sigma_1 \cup \{\epsilon\}) \times (\Gamma_{\alpha_1} \times \ldots \times \Gamma_{\alpha_k})$ into the finite subsets of $K_1 \times (I_{\alpha_1} \times \ldots \times I_{\alpha_k}) \times \Sigma_2^*$ such that

$$G_M = \{(\gamma_1, \ldots, \gamma_k)/\delta(q,a,(\gamma_1, \ldots, \gamma_k)) \neq \emptyset \text{ for some } q \text{ and } a\}$$

is finite.

$\mathcal{D}^o$ is said to be a multitape abstract family of transducers (abbreviated, multitape AFT).

In a multitape transducer, K_1, Σ_1, and Σ_2 are called the "states," "inputs," and "outputs," resp.

The notation for the movement of multitape transducers is similar to that for acceptors.

Notation. Let $M = (K_1, \Sigma_1, \Sigma_2, \delta, q_o, \upsilon)$ be a multitape transducer. Let $\vdash$ be the relation on $K_1 \times \Sigma_1^* \times (\Gamma_{\alpha_1} \times \ldots \times \Gamma_{\alpha_k}) \times \Sigma_2^*$ defined as follows:

For a in $\Sigma_1 \cup \{\epsilon\}$, w in Σ_1^*, and y' in Σ_2^*,

$$(q, aw, (\gamma_1, \ldots, \gamma_k), y') \vdash (q', w, (\gamma_1', \ldots, \gamma_k'), y'y)$$

if there exist $\bar{\gamma}_1, \ldots, \bar{\gamma}_k$, each $\bar{\gamma}_i$ in $g_{\alpha_i}(\gamma_i)$, such that $(q', (u_1, \ldots, u_k), y)$ is in $\delta(q, a, (\bar{\gamma}_1, \ldots, \bar{\gamma}_k))$ and $f_{\alpha_i}(\gamma_i, u_i) = \gamma_i'$ for each i.

The relations $\overset{n}{\vdash}$ and $\overset{*}{\vdash}$ are defined as in a multitape acceptor.

A multitape transducer realizes a function in the following way.

Notation. Let $M = (K_1, \Sigma_1, \Sigma_2, \delta, q_o, \upsilon)$ be a multitape transducer. For each w in Σ_1^* let

$$M(w) = \{z/(q_o, w, \epsilon, \epsilon) \overset{*}{\vdash} (p, \epsilon, \epsilon, z) \text{ for some } p \text{ in } K_1\}.$$

For each $L \subseteq \Sigma_1^*$, let $M(L) = \bigcup_{w \text{ in } L} M(w)$.

We shall need some special types of transducers.

Definition. Let $M = (K_1, \Sigma_1, \Sigma_2, \delta, q_o, \upsilon)$ be a multitape AFA.

(1) M is <u>ϵ-input bounded</u> if there exists $m \geq 0$ such that for all $q, q', \gamma_i, \gamma_i'$, y, and y', $(q, \epsilon, (\gamma_1, \ldots, \gamma_k), y) \overset{n}{\vdash} (q', \epsilon, (\gamma_1', \ldots, \gamma_k'), y')$ implies $n \leq m$.

(2) M is <u>ϵ-output bounded</u> if there exists $m \geq 0$ such that for all q, q', w, w', γ_i, and γ_i', $(q, w, (\gamma_1, \ldots, \gamma_k), \epsilon) \overset{n}{\vdash} (q', w', (\gamma_1', \ldots \gamma_k'), \epsilon)$ implies $n \leq m$.

(3) M is <u>partially ϵ-output bounded</u> if there exists $m \geq 0$ such that for all q, q', w, γ_1, and γ_1', $(q, w, (\gamma_1, \ldots, \gamma_k), \epsilon) \overset{*}{\vdash} (q', \epsilon, (\gamma_1', \ldots, \gamma_k'), \epsilon)$ implies $|w| \leq m$.

Note that M is ϵ-output bounded if it is ϵ-input bounded and partially ϵ-output bounded. Also, if M is partially ϵ-output bounded then it is ϵ-output bounded.

<u>Notation</u>. Let $(\Omega_1, \mathcal{D}_1), \ldots, (\Omega_n, \mathcal{D}_n)$ be multitape AFA and $(\Omega_1, \mathcal{D}_1^o), \ldots, (\Omega_n, \mathcal{D}_n)$ the corresponding AFT. Let $\hat{\mathfrak{M}}_{\mathcal{D}_1 \ldots \mathcal{D}_n}$, $\mathfrak{M}_{\mathcal{D}_1} \ldots \mathfrak{M}_{\mathcal{D}_n}$ and $\mathfrak{M}^t_{\mathcal{D}_1 \ldots \mathcal{D}_n}$ be the sets of mappings defined by

$$\hat{\mathfrak{M}}_{\mathcal{D}_1 \ldots \mathcal{D}_n} = \{M_n M_{n-1} \ldots M_1 / \text{each } M_i \text{ in } \mathcal{D}_i^o\},$$

$$\mathfrak{M}_{\mathcal{D}_1 \ldots \mathcal{D}_n} = \{M_n \ldots M_1 / \text{each } M_i \text{ in } \mathcal{D}_i^o \text{ and partially } \epsilon\text{-output bounded}\},$$

and $$\mathfrak{M}^t_{\mathcal{D}_1 \ldots \mathcal{D}_n} = \{M_n \ldots M_1 / \text{each } M_i \text{ in } \mathcal{D}_i^o, \ \epsilon\text{-input bounded, and } \epsilon\text{-output bounded}\}.$$

Let $$\hat{\mathfrak{M}}_{\mathcal{D}_1 \ldots \mathcal{D}_n}(\mathcal{L}) = \{f(L)/f \text{ in } \hat{\mathfrak{M}}_{\mathcal{D}_1 \ldots \mathcal{D}_n}, \ L \text{ in } \mathcal{L}\},$$

$$\mathfrak{M}_{\mathcal{D}_1 \ldots \mathcal{D}_n}(\mathcal{L}) = \{f(L)/f \text{ in } \mathfrak{M}_{\mathcal{D}_1 \ldots \mathcal{D}_n}, \ L \text{ in } \mathcal{L}\},$$

and $$\mathfrak{M}^t_{\mathcal{D}_1 \ldots \mathcal{D}_n}(\mathcal{L}) = \{f(L)/f \text{ in } \mathfrak{M}^t_{\mathcal{D}_1 \ldots \mathcal{D}_n}, \ L \text{ in } \mathcal{L}\}.$$

We now present two lemmas which play a role analogous to that of Lemmas 2.1 and 2.2.

<u>Lemma 3.1</u>. Let $\mathcal{L}$ be an AFL containing $\{\epsilon\}$ and $(\Omega, \mathcal{D})$ a multitape AFA. Then

$$\hat{\mathcal{H}}(\mathcal{L} \wedge \mathcal{L}(\mathcal{D})) \subseteq \hat{\mathfrak{M}}_{\mathcal{D}}(\mathcal{L}),$$

$$\mathcal{H}(\mathcal{L} \wedge \mathcal{L}(\mathcal{D})) \subseteq \mathfrak{M}_{\mathcal{D}}(\mathcal{L}),$$

and $$\mathcal{H}(\mathcal{L} \wedge \mathcal{L}^t(\mathcal{D})) \subseteq \mathfrak{M}^t_{\mathcal{D}}(\mathcal{L}).$$

<u>Proof</u>. Let L_1 be in $\mathcal{L}$ and L_2 in $\mathcal{L}(\mathcal{D})$. Let $L_2 = L(D)$ for $D = (K_1, \Sigma_1, \delta_1, q_0, F_1, \nu)$ in $\mathcal{D}$, $\nu = (\alpha_1, \ldots, \alpha_k)$. Let c and $\overline{f}$ be new symbols in Σ and K resp. Then $L_1 c$ is in $\mathcal{L}$ and $L_2 c$ is in $\mathcal{L}(\mathcal{D})$. If L_2 is in $\mathcal{L}^t(\mathcal{D})$, then $L_2 c$ is in $\mathcal{L}^t(\mathcal{D})$. Let h be a homomorphism from Σ_1^* into Σ_2^*.

Let M be the multitape acceptor $(K_1 \cup \{\overline{f}\}, \Sigma_1 \cup \{c\}, \Sigma_2, \delta_2, q_0, \nu)$, where $\nu = (\alpha_1, \ldots, \alpha_k)$ and δ_2 is defined as follows:

(1) Let $(q', (u_1, \ldots, u_k), h(a))$ be in $\delta_2(q, a, (\gamma_1, \ldots, \gamma_k))$ if $(q', (u_1, \ldots, u_k))$ is in $\delta_1(q, a, (\gamma_1, \ldots, \gamma_k))$.

(2) Let $(\overline{f}, (1_{\alpha_1}, \ldots, 1_{\alpha_k}), \epsilon)$ be in $\delta_2(p, c, (\epsilon, \ldots, \epsilon))$ for all p in F.

Then for w in Σ_1^*, q, q' in K_1,

$$(q, w, (\gamma_1, \ldots, \gamma_k), \epsilon) \overset{m}{\underset{M}{\vdash}} (q', \epsilon, (\gamma_1', \ldots, \gamma_k'), h(w))$$

if and only if

$$(q, w, (\gamma_1, \ldots, \gamma_k)) \overset{m}{\underset{D}{\vdash}} (q', \epsilon, (\gamma_1', \ldots, \gamma_k')).$$

Thus $M(L_1 c) = h(L_1 \cap L_2)$, so that $h(L_1 \cap L_2)$ is in $\hat{\mathfrak{M}}_{\mathcal{D}}(\mathcal{L})$. If h is ϵ-free, then M is partially ϵ-output bounded, so that $h(L_1 \cap L_2)$ is in $\mathfrak{M}_{\mathcal{D}}(\mathcal{L})$. If D is quasi-realtime and h is ϵ-free, then M is ϵ-input bounded and ϵ-output bounded, so that $h(L_1 \cap L_2)$ is in $\mathfrak{M}_{\mathcal{D}}^t(\mathcal{L})$. This completes the proof.

<u>Lemma 3.2</u>. Let $\mathcal{L}$ be an AFL containing $\{\epsilon\}$ and $(\Omega, \mathcal{D})$ a multitape AFA, with $(\Omega, \mathcal{D}^o)$ the corresponding AFT. Then

$$\hat{\mathfrak{M}}_{\mathcal{D}}(\mathcal{L}) \subseteq \hat{\mathcal{H}}(\mathcal{L} \wedge \mathcal{L}(\mathcal{D})),$$

$$\mathfrak{M}_{\mathcal{D}}(\mathcal{L}) \subseteq \mathcal{H}(\mathcal{L} \wedge \mathcal{L}(\mathcal{D})),$$

and $$\mathfrak{M}^t(\mathcal{L}) \subseteq \mathcal{H}(\mathcal{L} \wedge \mathcal{L}^t(\mathcal{D})).$$

Proof. Let L be in $\mathcal{L}$ and $M = (K_1,\Sigma_1,\Sigma_2,\delta_1,q_0,\upsilon)$ in $\mathcal{D}^0$, with $\upsilon = (\alpha_1, \ldots, \alpha_k)$. Let

$$n = \max\{|z|/(p',(u_1, \ldots, u_k),z) \text{ in } \delta(p,a,(\gamma_1, \ldots, \gamma_k)) \text{ for some } p, p',$$
$$a, u_1, \ldots, u_k, \gamma_1, \ldots, \gamma_k\}.$$

Since G_M is finite, n exists. For w in Σ_2^*, $1 \leq |w| \leq n$, let $\bar{w}$ be a new symbol and Σ_3 the set of all such $\bar{w}$. Let $D = (K_1 \cup (K \times \Sigma_3), \Sigma_1 \cup \Sigma_3, \delta_2, q_0, K_1, \upsilon)$, where δ is defined as follows:

(1) If $(q',(u_1, \ldots, u_k),z)$ is in $\delta_1(q,a,(\gamma_1, \ldots, \gamma_k))$, then

(α) $((q',\bar{z}),(u_1, \ldots, u_k))$ is in $\delta_2(q,a,(\gamma_1, \ldots, \gamma_k))$ if $z \neq \epsilon$.

(β) $(q',(u_1, \ldots, u_k))$ is in $\delta_2(q,a,(\gamma_1, \ldots, \gamma_k))$ if $z=\epsilon$.

(2) $(q,(1(\alpha_1,\gamma_1), \ldots, 1(\alpha_k,\gamma_k)))$ is in $\delta_2((q,\bar{z}),\bar{z},(\gamma_1, \ldots, \gamma_k))$ for all $(q,\bar{z})$ in $K_1 \times \Sigma_3$ and all $(\gamma_1, \ldots, \gamma_k)$ in G_M.

Let $L_1 = L(D)$ and $L_2 = \mathrm{Shuf}(L,\Sigma_3^*)$.(14) Then L_1 is in $\mathcal{L}(\mathcal{D})$ and L_2 is in $\mathcal{L}$.(15) Let h be the homomorphism on $\Sigma_1 \cup \Sigma_3$ defined by $h(a) = \epsilon$ for a in Σ_1 and $h(\bar{w}) = w$ for $\bar{w}$ in Σ_3. Then $M(L) = h(L_2 \cap L_1)$, so that $M(L)$ is in $\hat{\mathcal{H}}(\mathcal{L}\wedge\mathcal{L}(\mathcal{D}))$. Suppose M is partially ϵ-output bounded. Then there exists uch that $(q,w,(\gamma_1 \ldots \gamma_k),\epsilon) \overset{*}{\vdash} (q',\epsilon,(\gamma_1', \ldots, \gamma_k'),\epsilon)$ implies $|w| \leq m$. Then for any xwy in L_1, $h(w) = \epsilon$ implies $|w| \leq m+1$. Thus h is ϵ-limited on L_1 and so on $L_1 \cap L_2$. Now $\mathcal{H}(\mathcal{L}\wedge\mathcal{L}(\mathcal{D}))$ is an AFL. Since $\mathcal{L}$ contains $\{\epsilon\}$ and $\mathcal{L}(\mathcal{D})$ contains $\{\epsilon\}$, $\mathcal{H}(\mathcal{L}\wedge\mathcal{L}(\mathcal{D}))$ contains $\{\epsilon\}$. Therefore $h(L_1 \cap L_2)$ is in $\mathcal{H}(\mathcal{L}\wedge\mathcal{L}(\mathcal{D}))$ [5]. Hence $M(L)$ is in $\mathcal{H}(\mathcal{L}\wedge\mathcal{L}(\mathcal{D}))$ if M is partially ϵ-output bounded. If M is ϵ-input bounded,

(14) Let L_1 and L_2 be languages. Then $\mathrm{Shuf}(L_1,L_2)$, the shuffles of L_1 by L_2, is defined as the set

$$\{w_1y_1 \ldots w_ny_n/w_1 \ldots w_n \text{ in } L_1, y_1 \ldots y_n \text{ in } L_2, n \geq 1\}.$$

(15) It is known [5] that if $\mathcal{L}$ is an AFL, L is in $\mathcal{L}$ and R is regular, then $\mathrm{Shuf}(L,R)$ is in $\mathcal{L}$.

then obviously D is quasi-realtime and L(D) is in $\mathcal{L}^t(\mathcal{D})$. Thus, if M is ϵ-input bounded and ϵ-output bounded, and thus partially ϵ-output bounded, then $h(L_2 \cap L_1) = M(L)$ is in $\mathcal{H}(\mathcal{L} \wedge \mathcal{L}^t(\mathcal{D}))$. This completes the proof.

Using the two previous lemmas we now derive

Theorem 3.1. Let $(\Omega_1, \mathcal{D}_1), \ldots, (\Omega_n, \mathcal{D}_n)$ be multitape AFA and $\mathcal{L}$ an AFL containing $\{\epsilon\}$. Then

$$\hat{\mathfrak{M}}_{\mathcal{D}_1 \ldots \mathcal{D}_n}(\mathcal{L}) = \hat{\mathcal{H}}(\mathcal{L} \wedge \mathcal{L}(\mathcal{D}_1) \wedge \ldots \wedge \mathcal{L}(\mathcal{D}_n)),$$

$$\mathfrak{M}_{\mathcal{D}_1 \ldots \mathcal{D}_n}(\mathcal{L}) = \mathcal{H}(\mathcal{L} \wedge \mathcal{L}(\mathcal{D}_1) \wedge \ldots \wedge \mathcal{L}(\mathcal{D}_n)),$$

and $$\mathfrak{M}^t_{\mathcal{D}_1 \ldots \mathcal{D}_n}(\mathcal{L}) = \mathcal{H}(\mathcal{L} \wedge \mathcal{L}^t(\mathcal{D}_1) \wedge \ldots \wedge \mathcal{L}^t(\mathcal{D}_n)).$$

Proof. We proceed by induction on n. The result holds for n = 1 by the previous lemmas. Suppose $n \geq 2$ and the theorem is true for n-1. Then

$$\begin{aligned}
\hat{\mathfrak{M}}_{\mathcal{D}_1 \ldots \mathcal{D}_n}(\mathcal{L}) &= \hat{\mathfrak{M}}_{\mathcal{D}_n}(\mathfrak{M}_{\mathcal{D}_1 \ldots \mathcal{D}_{n-1}}(\mathcal{L})), \text{ by definition,} \\
&= \hat{\mathfrak{M}}_{\mathcal{D}_n}(\hat{\mathcal{H}}(\mathcal{L} \wedge \mathcal{L}(\mathcal{D}_1) \wedge \ldots \wedge \mathcal{L}(\mathcal{D}_{n-1})), \text{ by induction,} \\
&= \hat{\mathcal{H}}(\hat{\mathcal{H}}(\mathcal{L} \wedge \mathcal{L}(\mathcal{D}_1) \wedge \ldots \wedge \mathcal{L}(\mathcal{D}_{n-1})) \wedge \mathcal{L}(\mathcal{D}_n)), \text{ by induction.}
\end{aligned}$$

Now $$\begin{aligned}
&\hat{\mathcal{H}}[\hat{\mathcal{H}}(\mathcal{L} \wedge \mathcal{L}(\mathcal{D}_1) \wedge \ldots \wedge \mathcal{L}(\mathcal{D}_{n-1})) \wedge \mathcal{L}(\mathcal{D}_n))\\
&\supseteq \hat{\mathcal{H}}[\mathcal{L} \wedge \mathcal{L}(\mathcal{D}_1) \wedge \ldots \wedge \mathcal{L}(\mathcal{D}_n)]\\
&= \hat{\mathcal{H}}\hat{\mathcal{H}}[\mathcal{L} \wedge \mathcal{L}(\mathcal{D}_1) \wedge \ldots \wedge \mathcal{L}(\mathcal{D}_n)]\\
&\supseteq \hat{\mathcal{H}}[\hat{\mathcal{H}}(\mathcal{L} \wedge \mathcal{L}(\mathcal{D}_1) \wedge \ldots \wedge \mathcal{L}(\mathcal{D}_{n-1})) \wedge \hat{\mathcal{H}}(\mathcal{L}(\mathcal{D}_n))], \text{ by Theorem 1.2 c,}\\
&= \hat{\mathcal{H}}[\hat{\mathcal{H}}(\mathcal{L} \wedge \mathcal{L}(\mathcal{D}_1) \wedge \ldots \wedge \mathcal{L}(\mathcal{D}_{n-1})) \wedge \mathcal{L}(\mathcal{D}_n)].
\end{aligned}$$

Thus $\hat{\mathfrak{m}}_{\mathcal{A}_1 \ldots \mathcal{A}_n}(\mathcal{L}) = \hat{\mathcal{H}}[\hat{\mathcal{H}}(\mathcal{L}\wedge\mathcal{L}(\mathcal{A}_1) \wedge \ldots \wedge \mathcal{L}(\mathcal{A}_{n-1}))\wedge\mathcal{L}(\mathcal{A}_n))$

$= \hat{\mathcal{H}}[\mathcal{L}\wedge\mathcal{L}(\mathcal{A}_1) \wedge \ldots \wedge \mathcal{L}(\mathcal{A}_n)].$

In a similar manner,

$$\mathfrak{m}_{\mathcal{A}_1 \ldots \mathcal{A}_n}(\mathcal{L}) = \mathfrak{m}_{\mathcal{A}_n}(\mathfrak{m}_{\mathcal{A}_1 \ldots \mathcal{A}_{n-1}}(\mathcal{L}))$$

$$= \mathfrak{m}_{\mathcal{A}_n}[\mathcal{H}(\mathcal{L}\wedge\mathcal{L}(\mathcal{A}_1) \wedge \ldots \wedge \mathcal{L}(\mathcal{A}_{n-1})]$$

$$= \mathcal{H}[\mathcal{H}(\mathcal{L}\wedge\mathcal{L}(\mathcal{A}_1) \wedge \ldots \wedge \mathcal{L}(\mathcal{A}_{n-1}))\wedge\mathcal{L}(\mathcal{A}_n)]$$

$$= \mathcal{H}[\mathcal{L}\wedge\mathcal{L}(\mathcal{A}_1) \wedge \ldots \wedge \mathcal{L}(\mathcal{A}_n)].$$

Similarly $\mathfrak{m}^t_{\mathcal{A}_1 \ldots \mathcal{A}_n}(\mathcal{L}) = \mathcal{H}[\mathcal{L}\wedge\mathcal{L}^t(\mathcal{A}_1) \wedge \ldots \wedge \mathcal{L}^t(\mathcal{A}_n)].$

As a corollary, we get

<u>Theorem 3.2</u>. Let $(\Omega_1,\mathcal{A}_1), \ldots, (\Omega_n,\mathcal{A}_n)$ be single-tape AFA and $\mathcal{L}$ an AFL containing $\{\epsilon\}$. Then $\hat{\mathfrak{m}}_{\mathcal{A}_1 \wedge \ldots \wedge \mathcal{A}_n}(\mathcal{L}) = \hat{\mathfrak{m}}_{\mathcal{A}_1 \ldots \mathcal{A}_n}(\mathcal{L})$ and $\mathfrak{m}^t_{\mathcal{A}_1 \wedge \ldots \wedge \mathcal{A}_n}(\mathcal{L}) = \mathfrak{m}^t_{\mathcal{A}_1 \ldots \mathcal{A}_n}(\mathcal{L})$.

<u>Proof</u>. $\hat{\mathfrak{m}}_{\mathcal{A}_1 \wedge \ldots \wedge \mathcal{A}_n}(\mathcal{L}) = \hat{\mathcal{H}}(\mathcal{L}\wedge\mathcal{L}(\mathcal{A}_1 \wedge \ldots \wedge \mathcal{A}_n))$, by Theorem 3.1,

$= \hat{\mathcal{H}}(\mathcal{L}\wedge\hat{\mathcal{H}}(\mathcal{L}(\mathcal{A}_1 \wedge \ldots \wedge \mathcal{L}(\mathcal{A}_n))$, by Theorem 2.1,

$= \hat{\mathcal{H}}(\mathcal{L}\wedge\mathcal{L}(\mathcal{A}_1) \wedge \ldots \wedge \mathcal{L}(\mathcal{A}_n))$, as shown in the proof of Theorem 3.1,

$= \hat{\mathfrak{m}}_{\mathcal{A}_1 \ldots \mathcal{A}_n}(\mathcal{L})$, by Theorem 3.1.

The proof that $\mathfrak{m}^t_{\mathcal{A}_1 \wedge \ldots \wedge \mathcal{A}_n}(\mathcal{L}) = \mathfrak{m}^t_{\mathcal{A}_1 \ldots \mathcal{A}_n}(\mathcal{L})$ follows similarly.

Remark. Theorem 3.2 asserts that the composition of single-tape transducers is equivalent to a multitape transducer, from the point of view of families of sets produced as output by (1) all transducers, and (2) ϵ-input bounded and ϵ-output bounded transducers. In general, however, $\mathfrak{M}_{\mathcal{D}_1 \ldots \mathcal{D}_n}(\mathcal{L}) \neq \mathfrak{M}_{\mathcal{D}_1 \wedge \ldots \wedge \mathcal{D}_n}(\mathcal{L})$. For, let $\mathcal{L}$ be the family of regular sets and $\mathcal{D}_1 = \mathcal{D}_2 = \mathcal{D}_p$, the family of pda. Then

$$\mathfrak{M}_{\mathcal{D}_p \wedge \mathcal{D}_p}(\mathcal{L}) = \mathcal{H}(\mathcal{L} \wedge \mathcal{L}(\mathcal{D}_p \wedge \mathcal{D}_p)) = \mathcal{L}(\mathcal{D}_p \wedge \mathcal{D}_p),$$

which is the family of all r.e. sets. Since $\mathcal{L}(\mathcal{D}_p) = \mathcal{L}^t(\mathcal{D}_p)$ [14],

$$\begin{aligned}\mathfrak{M}_{\mathcal{D}_p \mathcal{D}_p}(\mathcal{L}) &= \mathcal{H}(\mathcal{L} \wedge \mathcal{L}(\mathcal{D}_p) \wedge \mathcal{L}(\mathcal{D}_p)) \\ &= \mathcal{H}(\mathcal{L}(\mathcal{D}_p) \wedge \mathcal{L}(\mathcal{D}_p)) \\ &= \mathcal{H}(\mathcal{L}^t(\mathcal{D}_p) \wedge \mathcal{L}^t(\mathcal{D}_p)).\end{aligned}$$

Now $\mathcal{H}(\mathcal{L}^t(\mathcal{D}_p) \wedge \mathcal{L}^t(\mathcal{D}_p))$ contains only recursive sets (in fact, only context-sensitive languages and context-sensitive languages union $\{\epsilon\}$). Thus $\mathfrak{M}_{\mathcal{D}_p \mathcal{D}_p}(\mathcal{L})$ is a proper subfamily of $\mathfrak{M}_{\mathcal{D}_p \wedge \mathcal{D}_p}(\mathcal{L})$.

Example. The list languages have already been characterized as each of the following families:

(1) The ϵ-free sets which are recognized by quasi-realtime 2-pda acceptors.

(2) The ϵ-free sets which are the ϵ-free homomorphic image of the intersection of quasi-realtime pda languages, i.e., the ϵ-free sets in $\mathcal{H}(\mathcal{L}^t(\mathcal{D}_p) \wedge \mathcal{L}^t(\mathcal{D}_p))$.

(3) The sets obtained from context-free control sets acting on ϵ-free context-free grammars.

Using the previous theorems we may add the following characterizations:

(4) The ϵ-free sets obtained from partially ϵ-output bounded pushdown transducers operating on context-free languages, i.e., the ϵ-free sets in $\mathfrak{M}_{\mathcal{S}_p}(\mathcal{L}^t(\mathcal{S}_p)) = \mathcal{H}(\mathcal{L}^t(\mathcal{S}_p)\wedge\mathcal{L}(\mathcal{S}_p)) = \mathcal{H}(\mathcal{L}^t(\mathcal{S}_p)\wedge\mathcal{L}^t(\mathcal{S}_p))$.

(5) The ϵ-free sets obtained from ϵ-output bounded and ϵ-input bounded pushdown transducers operating on context-free languages, i.e., the ϵ-free sets in $\mathfrak{M}^t_{\mathcal{S}_p}(\mathcal{L}^t(\mathcal{S}_p)) = \mathcal{H}(\mathcal{L}^t(\mathcal{S}_p)\wedge\mathcal{L}^t(\mathcal{S}_p))$.

(6) The ϵ-free sets obtained from ϵ-output-free pushdown transducers operating on context-free languages. [For, let $\mathcal{F}$ be the set obtained from ϵ-output-free pushdown transducers acting on context-free languages. It can be shown that $\mathfrak{M}^t_{\mathcal{S}_p}(\mathcal{L}^t(\mathcal{S}_p)) \subseteq \mathcal{F} \subseteq \mathfrak{M}_{\mathcal{S}_p}(\mathcal{L}^t(\mathcal{S}_p))$, the second containment by a recoding argument. Since $\mathfrak{M}_{\mathcal{S}_p}(\mathcal{L}^t(\mathcal{S}_p)) = \mathcal{H}(\mathcal{L}^t(\mathcal{S}_p\wedge\mathcal{L}^t(\mathcal{S}_p)) = \mathfrak{M}^t_{\mathcal{S}_p}(\mathcal{L}^t(\mathcal{S}_p))$, $\mathcal{F} = \mathfrak{M}_{\mathcal{S}_p}(\mathcal{L}^t(\mathcal{S}_p))$.

Section 4. Nested Multitape AFA

In this section we study "nested" multitape AFA. We shall see that they allow a representation of the substitution of AFL into AFL.

Intuitively, a multitape acceptor is "nested" if each move can change at most one storage tape, and all tapes to the right of this one are ϵ. In order to express these two conditions in our formalism we need to distinguish

identity instructions in our acceptors. More precisely, we have

<u>Notation</u>. For each α and γ, let $\psi_\alpha(\gamma) = \{u \text{ in } I_\alpha / f_\alpha(\gamma', u) = \gamma' \text{ for all } \gamma' \text{ in } g_\alpha^{-1}(\gamma)\}$, where $g_\alpha^{-1}(\gamma) = \{\gamma'' / \gamma \text{ in } g_\alpha(\gamma'')\}$.

By definition of an AFA schema, $\psi_\alpha(\gamma) \neq \emptyset$ for each γ in $g_\alpha(\Gamma_\alpha^*)$.

We are now able to define a "nested" multitape AFA.

<u>Definition</u>. A <u>nested multitape AFA</u> is a pair $(\Omega, \mathcal{D}^N)$, where

(1) $(\Omega, \mathcal{D})$ is a multitape AFA, with $\Omega = (K, \Sigma, \mathcal{A}, <, \mu)$,

and (2) $\mathcal{D}^N$ is the set of all $D = (K_1, \Sigma_1, \delta, q_o, F, (\alpha_1, \ldots, \alpha_k))$ in $\mathcal{D}$ with the following property (for arbitrary q', q in K_1, a in $\Sigma_1 \cup \{\epsilon\}$, $(u_1, \ldots, u_k)$ in $I_{\alpha_1} \times \ldots \times I_{\alpha_k}$, and $(\gamma_1, \ldots, \gamma_k)$ in G_D): If $(q', (u_1, \ldots, u_k))$ is in $\delta(q, a, (\gamma_1, \ldots, \gamma_k))$ and u_ℓ is not in $\psi_{\alpha_\ell}(\gamma_\ell)$ for some ℓ, then u_i is in $\psi_{\alpha_i}(\gamma_i)$ for all $i \neq \ell$ and $\gamma_i = \epsilon$ for all $i > \ell$.

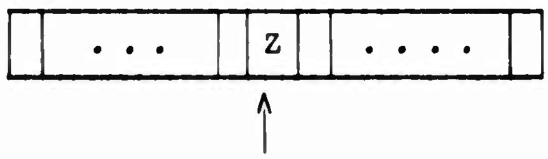
Each D in $\mathcal{D}^N$ is called a <u>nested</u> acceptor.

<u>Notation</u>. Let $\mathcal{L}(\mathcal{D}^N) = \{L(D)/D \text{ in } \mathcal{D}^N\}$ and $\mathcal{L}^t(\mathcal{D}^N) = \{L(D)/D \text{ in } \mathcal{D}^N$ and D quasi-realtime$\}$.

Note that if D is nested and $(q, a, (\gamma_1, \ldots, \gamma_k)) \vdash (q', \epsilon, (\gamma_1', \ldots, \gamma_k'))$, then there is at most one ℓ with $\gamma_\ell \neq \gamma_\ell'$, <u>and</u> either $\ell = k$ or $\gamma_i = \epsilon = \gamma_i'$ for $i > \ell$. Thus at most one tape of D is changed and all tapes to the right of it are inactive, i.e., are ϵ.

The meaning of the term "nested" becomes clearer if we consider some familiar AFA. Suppose an acceptor such as a pda, a Turing acceptor, or a one-way stack acceptor [6] has the storage configuration depicted in Figure 1,

that is, a tape with a read-write head which affects exactly one symbol of the storage tape. This type of configuration is usually reflected in the



<u>Figure 1</u>

formalism by a pointer symbol, say $\uparrow$, and the definition $g(xZ\uparrow y) = Z$, where Z is a symbol and x and y are words which may have other restrictions. (In a pda, $y = \epsilon$; see Example 4 in [5] for the definition of a one-way stack acceptor.) In these cases, activating a tape is equivalent to inserting a new tape, initially ϵ, enclosed in markers--say matched brackets--where the read-write head is. The nesting condition says that the head cannot leave the bracketed tape until the bracketed tape becomes ϵ. This is equivalent to preventing the multitape AFA from changing a tape until all tapes to the right are inactive. Restricting the device to n tapes is equivalent to restricting the depth of the nesting of brackets to n. For pda and Turing acceptors, nesting does not affect the computational power of the type of device. For one-way stack acceptors, we shall see later (Example 3) that nesting increases the computational power. We shall show in this section that nesting of devices is related to substitution in languages.

From the definition, it is clear that $\mathcal{L} = \mathcal{L}^N$ for each single-tape AFA. However, in general a nested multitape AFA need not be a multitape AFA as

defined in Section 1. To motivate the use of the word "AFA: after "nested multitape," we now show that a nested multitape AFA is equivalent (from the sets accepted point of view) to a single-tape AFA.

<u>Lemma 4.1.</u> For each nested multitape AFA $(\Omega,\mathcal{D}^N)$, there exists a single-tape AFA $(\overline{\Omega},\overline{\mathcal{D}})$ such that $\mathcal{L}^t(\overline{\mathcal{D}}) = \mathcal{L}^t(\mathcal{D}^N)$ and $\mathcal{L}(\overline{\mathcal{D}}) = \mathcal{L}(\mathcal{D}^N)$.

<u>Proof.</u> Let $\xi,\sigma(\upsilon)$, $\overline{\Gamma}$ and $\overline{g}$ be as in the proof of Lemma 1.1. For $\upsilon = (\alpha_1, \ldots, \alpha_k)$, $1 \le \ell \le k$, and u in I_{α_ℓ}, let $\sigma(\upsilon,\ell,u)$ be a new symbol and

$$\overline{I} = \{\sigma(\upsilon,\ell,u)/\upsilon = (\alpha_1, \ldots, \alpha_k),\ 1 \le \ell \le k,\ u \text{ in } I_{\alpha_\ell}\}\cup\{\sigma(\upsilon)/\text{all } \upsilon\}\cup\{\epsilon\}.$$

For $\upsilon = (\alpha_1, \ldots, \alpha_k)$, let $\overline{f}(\epsilon,\sigma(\upsilon)) = \sigma(\upsilon)\xi^{k+1}$, $\overline{f}(\sigma(\upsilon)\xi^{k+1},\epsilon) = \epsilon$, and $\overline{f}(\epsilon,\epsilon) = \epsilon$. For $\upsilon = (\alpha_1, \ldots, \alpha_k)$, $(x_1, \ldots, x_k)$ in $\Gamma^*_{\alpha_1} \times \ldots \times \Gamma^*_{\alpha_k}$, ℓ such that either $\ell = k$ or $x_i = \epsilon$ for all $i>\ell$, and u in I_{α_ℓ}, let

$$\overline{f}(\sigma(\upsilon)\xi x_1\xi \ldots x_k\xi,\sigma(\upsilon,\ell,u)) = \sigma(\upsilon)\xi x_1'\xi \ldots x_k'\xi,$$

where $x_i' = x_i$ for all i, $i\neq\ell$, and $x_\ell' = f_{\alpha_\ell}(x_\ell,u)$. By the same reasoning as in Lemma 1.1, $(\overline{\Gamma},\overline{I},\overline{f},\overline{g})$ is an AFA-schema. Let $\overline{\Omega} = (K,\Sigma,\overline{\Gamma},\overline{I},\overline{f},\overline{g})$.

Let $D = (K_1,\Sigma_1,\delta,q_0,F,\upsilon)$ be in $\mathcal{D}^N$. Let $\overline{q}_0$ and r_0 be new symbols in K and $\overline{D} = (K_1\cup\{\overline{q}_0,r_0\},\Sigma_1,\overline{\delta},\overline{q}_0,\{r_0\})$, where $\overline{\delta}$ is defined as follows:

(1) $\overline{\delta}(\overline{q}_0,\epsilon,\epsilon) = \{(q_0,\sigma(\upsilon))\}$.

(2) (r_0,ϵ) is in $\overline{\delta}(p,\epsilon,\sigma(\upsilon)\xi^{k+1})$ for each p in F.

(3) If $(q',(u_1, \ldots, u_k))$ is in $\delta(q,a,(\gamma_1, \ldots, \gamma_k))$, then $\overline{\delta}(q,a,\sigma(\upsilon)\xi\gamma_1\xi \ldots \gamma_k\xi)$ contains

(a) $(q',\sigma(\upsilon,1,u_1))$ if $\gamma_i = \epsilon$ and u_i is in $\psi_{\alpha_i}(\epsilon)$ for all i, $1 \le i \le k$.

(b) $(q',\sigma(\upsilon,\ell,u_\ell))$ if $\gamma_\ell \ne \epsilon$, $\gamma_i = \epsilon$ for all $i > \ell$, and u_i is in $\psi_{\alpha_i}(\gamma_i)$ for $1 \le i \le k$.

(c) $(q',\sigma(\upsilon,\ell,u_\ell))$ if u_ℓ is not in $\psi_{\alpha_\ell}(\gamma_\ell)$, u_i is in $\psi_{\alpha_i}(\gamma_i)$ for all $i \ne \ell$, and $\gamma_i = \epsilon$ for all $i > \ell$.

Clearly $L(D) = L(\overline{D})$ and $\overline{D}$ is quasi-realtime if D is.

Now let $\overline{D} = (K_1,\Sigma_1,\delta,q_0,F)$ be in $\mathcal{S}$. Let

$S = \{\upsilon / (q',\sigma(\upsilon))$ in $\delta(q,a,\epsilon)$ for some q and a$\}$.

As in the proof of Lemma 1.1, we may assume that there exists $\upsilon_0=(\alpha_1, \ldots, \alpha_n)$ such that if υ is in S then $\upsilon = (\alpha_{j_1}, \ldots, \alpha_{j_k})$ for some $1 \le j_1 < \ldots < j_k \le n$.

We may also assume that if $(q',\sigma(\upsilon_1,\ell,u))$ is in $\delta(q,a,\sigma(\upsilon_2)\xi\gamma_1\xi \ldots \gamma_k\xi)$, then $\upsilon_1 = \upsilon_2 = (\alpha_{j_1}, \ldots, \alpha_{j_k})$ is in S, u is in $I_{\alpha_{j_\ell}}$, γ_i is in $g_{\alpha_{j_i}}(\Gamma^*_{\alpha_{j_i}})$ for $1 \le i \le k$, and $\gamma_i = \epsilon$ for $i > \ell$. Let $D = (K_1 \times (S \cup \{\epsilon\}), \Sigma_1, \delta_1, (q_0,\epsilon), F \times \{\epsilon\}, \upsilon_0)$, where δ_1 is defined as follows (for arbitrary $\upsilon = (\alpha_{j_1}, \ldots, \alpha_{j_k})$):

(4) $((q',\epsilon), (1_{\alpha_1}, \ldots, 1_{\alpha_n}))$ is in $\delta_1((q,\epsilon),a,(\epsilon, \ldots, \epsilon))$ if (q',ϵ) is in $\delta(q,a,\epsilon)$.

(5) $((q',\epsilon), (1_{\alpha_1}, \ldots, 1_{\alpha_n}))$ is in $\delta_1((q,\upsilon),a,(\epsilon, \ldots, \epsilon))$ if (q',ϵ) is in $\delta(q,a,\sigma(\upsilon)\xi^{k+1})$.

(6) $((q',\nu), (1_{\alpha_1}, \ldots, 1_{\alpha_n}))$ is in $\delta_1((q,\epsilon),a,(\epsilon, \ldots, \epsilon))$ if $(q',\sigma(\nu))$ is in $\delta(q,a,\epsilon)$.

(7) $((q',\nu), (u_1', \ldots, u_n'))$ is in $\delta_1((q,\nu),a,(\gamma_1', \ldots, \gamma_n'))$ if $(q',\sigma(\nu,\ell,u))$ is in $\delta(q,a,\sigma(\nu)\xi\gamma_{j_1}\xi \ldots \gamma_{j_k}\xi)$, where $\gamma_{j_i} = \epsilon$ for all $i>\ell$, $u_{j_i}' = 1(\alpha_{j_i}, \gamma_{j_i})$ for all $i\neq\ell$, $u_{j_\ell}' = u$, and $\gamma_j' = \epsilon$ and $u_j' = 1_{\alpha_j}$ for all j not in $\{j_1, \ldots, j_k\}$.

Then $L(D) = L(\overline{D})$ and D is quasi-realtime if $\overline{D}$ is.

From Lemma 4.1 there immediately follows

Theorem 4.1. For each nested multitape AFA $\mathcal{D}^N$, $\mathcal{L}(\mathcal{D}^N)$ is a full AFL and $\mathcal{L}^t(\mathcal{D}^N)$ is an AFL containing $\{\epsilon\}$.

We now present some definitions and remarks about substitution, the operation to be associated with nesting.

Definition. Let $L \subseteq \Sigma_1^*$ and for each a in Σ_1 let $L_a \subseteq \Sigma_a^*$. Let τ be the function defined on Σ_1^* by $\tau(\epsilon) = \{\epsilon\}$, $\tau(a) = L_a$ for each a in Σ_1, and $\tau(a_1 \ldots a_n) = \tau(a_1) \ldots \tau(a_n)$ for each a_i in Σ_1 and $k\geq 1$. Then τ is called a substitution. τ is extended to $2^{\Sigma_1^*}$ by defining $\tau(X) = \bigcup_{x \text{ in } X} \tau(x)$ for all $X \subseteq \Sigma_1^*$. If $\tau(a) \subseteq \Sigma_a^+$ for each a in Σ_1, then τ is called ϵ-free.

Notation. Let $\mathcal{L}_1$ and $\mathcal{L}_2$ be families of languages. Let $\hat{\sigma}(\mathcal{L}_1,\mathcal{L}_2)$ $(\sigma(\mathcal{L}_1,\mathcal{L}_2))$ be the family of all sets $\tau(L_1)$, where $L_1 \subseteq \Sigma_1^*$ is in $\mathcal{L}_1$ and τ is a (ϵ-free) substitution such that $\tau(a)$ is in $\mathcal{L}_2$ for each a in Σ_1.

We usually write $\sigma(\mathcal{L}_1,\mathcal{L}_2)$ as $\mathcal{L}_1 \sigma \mathcal{L}_2$ and $\hat{\sigma}(\mathcal{L}_1,\mathcal{L}_2)$ as $\mathcal{L}_1 \hat{\sigma} \mathcal{L}_2$.

<u>Remarks</u>. (1) Let $\mathcal{L}_1$ and $\mathcal{L}_2$ be families of languages. Let $\mathcal{I}_2 = \{L \text{ in } \mathcal{L}_2 / \epsilon \text{ not in } L\}$. Then $\mathcal{L}_1\sigma \mathcal{L}_2 = \mathcal{L}_1\hat{\sigma} \mathcal{I}_2$. Note that $\mathcal{I}_2$ is an AFL if $\mathcal{L}_2$ is.

(2) It was shown in [13] that $\hat{\sigma}$ is associative on families of languages closed under isomorphism, i.e., $(\mathcal{L}_1\hat{\sigma} \mathcal{L}_2)\hat{\sigma} \mathcal{L}_3 = \mathcal{L}_1\hat{\sigma}(\mathcal{L}_2\sigma \mathcal{L}_3)$ if $\mathcal{L}_1,\mathcal{L}_2$, and $\mathcal{L}_3$ are families of languages closed under isomorphism. The same proof shows that σ is associative on such families of languages. Because of this associativity, we shall omit the parentheses in iterated applications of σ, resp. $\hat{\sigma}$, when the underlying families are closed under isomorphism, as in AFL.

(3) Neither σ nor $\hat{\sigma}$ is commutative, even on AFL, i.e., both $\mathcal{L}_1\sigma \mathcal{L}_2 = \mathcal{L}_2\sigma \mathcal{L}_1$ and $\mathcal{L}_1\hat{\sigma} \mathcal{L}_2 = \mathcal{L}_2\hat{\sigma} \mathcal{L}_1$ are false for AFL $\mathcal{L}_1$ and $\mathcal{L}_2$. For let $\mathcal{L}_1$ be the quasi-realtime one-way stack languages and $\mathcal{L}_2$ the context-free languages. Then $\mathcal{L}_2\sigma \mathcal{L}_1 \subseteq \mathcal{L}_1$, but $\mathcal{L}_1\sigma \mathcal{L}_2 \subseteq \mathcal{L}_1$ is false [16]. The situation for $\hat{\sigma}$ follows from that for σ by Remark 1. [A separate example for $\hat{\sigma}$ is to let $\mathcal{L}_1$ be the recursive sets and $\mathcal{L}_2$ the regular sets. Then $\mathcal{L}_2\hat{\sigma} \mathcal{L}_1 = \mathcal{L}_1$, but $\mathcal{L}_1\sigma \mathcal{L}_2$ is the family of r.e. sets and thus not $\mathcal{L}_1$.

(4) If $\mathcal{L}_1$ and $\mathcal{L}_2$ are AFL, with $\mathcal{L}_1$ full, then $\mathcal{L}_1\sigma \mathcal{L}_2 = \mathcal{L}_1\hat{\sigma} \mathcal{L}_2$

<u>Proof</u>. Clearly $\mathcal{L}_1\sigma \mathcal{L}_2 \subseteq \mathcal{L}_1\hat{\sigma} \mathcal{L}_2$. To see the reverse containment, let $L_1 \subseteq \Sigma_1^*$ be in $\mathcal{L}_1$ and τ a substitution such that $\tau(a)$ is in $\mathcal{L}_2$ for all a in Σ_1. Let τ_1 be the substitution on Σ_1^* defined by $\tau_1(a) = \{a\}$ if ϵ is not in $\tau(a)$ and $\tau_1(a) = \{a,\epsilon\}$ if ϵ is in $\tau(a)$. Since $\mathcal{L}_1$ is a full AFL, $\tau_1(L_1)$ is in $\mathcal{L}_1$. Let

τ_2 be the substitution on Σ_1^* defined by $\tau_2(a) = \tau(a)-\{\epsilon\}$ for each a. Then $\tau_2(a)$ is in $\mathcal{L}_2$ for each a. Thus $\tau(L_1) = \tau_2(\tau_1(L_1))$ is in $\mathcal{L}_1\sigma\ \mathcal{L}_2$, so that $\mathcal{L}_1\hat{\sigma}\ \mathcal{L}_2 \subseteq \mathcal{L}_1\sigma\ \mathcal{L}_2$.

(5) If $\mathcal{L}_1$ and $\mathcal{L}_2$ are AFL, with $\mathcal{L}_2$ containing $\{\epsilon\}$, then

$$\mathcal{L}_1\hat{\sigma}\ \mathcal{L}_2 = \hat{\mathcal{H}}(\mathcal{L}_1)\ \hat{\sigma}\ \mathcal{L}_2 = \hat{\mathcal{H}}(\mathcal{L}_1)\ \sigma\ \mathcal{L}_2.$$

Proof. Since $\hat{\mathcal{H}}(\mathcal{L}_1)$ is a full AFL, $\hat{\mathcal{H}}(\mathcal{L}_1)\ \hat{\sigma}\ \mathcal{L}_2 = \hat{\mathcal{H}}(\mathcal{L}_1)\ \sigma\ \mathcal{L}_2$ by Remark 4. Obviously $\mathcal{L}_1\hat{\sigma}\ \mathcal{L}_2 \subseteq \hat{\mathcal{H}}(\mathcal{L}_1)\ \hat{\sigma}\ \mathcal{L}_2$. To see the reverse inequality, let L_1 be in $\mathcal{L}_1$, $L_1 \subseteq \Sigma_1^*$, h a homomorphism of Σ_1^* into Σ_2^*, and τ a substitution on Σ_2^* such that $\tau(a)$ is in $\mathcal{L}_2$ for each a in Σ_2. Let c be a new symbol and $\bar{\tau}$ the substitution on $(\Sigma_2\cup\{c\})^*$ defined by $\bar{\tau}(a) = \tau(a)$ for each a in Σ_2 and $\bar{\tau}(c) = \{\epsilon\}$. Let $\bar{h}$ be the homomorphism on Σ_1^* defined by (i) $\bar{h}(a) = h(a)$ if a is in Σ_1 and $h(a)\neq a$, and (ii) $\bar{h}(a) = c$ if a in Σ_1 and $h(a) = \epsilon$. Since $\bar{h}$ is ϵ-free, $\bar{h}(L_1)$ is in $\mathcal{L}_1$. Clearly $\tau(h(L_1)) = \bar{\tau}(\bar{h}(L_1))$ is in $\mathcal{L}_1\hat{\sigma}\ \mathcal{L}_2$. Thus $\hat{\mathcal{H}}(\mathcal{L}_1)\hat{\sigma}\ \mathcal{L}_2 \subseteq \mathcal{L}_1\hat{\sigma}\ \mathcal{L}_2$ and the proof is complete.

We now present two lemmas that play the roles of Lemmas 2.1 and 2.2 of Section 2.

Lemma 4.2. Let $\mathcal{D}_1$ and $\mathcal{D}_2$ be multitape AFA and $\mathcal{D}_3 = \mathcal{D}_1\wedge\mathcal{D}_2$. Then

$$\mathcal{L}^t(\mathcal{D}_1^N)\ \sigma\ \mathcal{L}^t(\mathcal{D}_2^N) \subseteq \mathcal{L}^t(\mathcal{D}_3^N)$$

and $\mathcal{L}(\mathcal{D}_1^N)\ \sigma\ \mathcal{L}(\mathcal{D}_2^N) \subseteq \mathcal{L}(\mathcal{D}_3^N)$.

Proof. Let $\Omega_i = (K,\Sigma,\mathcal{C}_i,<_i,\mu_i)$ for $i = 1,2$. Let $D_1 = (K_1,\Sigma_1,\delta_1,q_0,F_1,\nu_1)$ be in $\mathcal{D}_1^N$ and for each a in Σ_1 let $D_a = (K_a,\Sigma_a,\delta_a,q_a,F_a,\nu_a)$ be in $\mathcal{D}_2^N$. We may

assume that $K_1 \cap K_a = K_a \cap K_b$ for all a and b, $a \neq b$, in Σ_1. By extending each ν_a if necessary, we may assume that there exists $\nu_2 = (\beta_1, \ldots, \beta_\ell)$ such that $\nu_a = \nu_2$ for all a in Σ_1. Let τ be the substitution on Σ_1^* defined by $\tau(a) = L(D_a)$ for each a in Σ_1. We shall construct D_3 in $\mathfrak{D}_3^N$ such that $L(D_3) = \tau(L(D_1))$.

Let $D_3 = (K_3, \bigcup_{a \text{ in } \Sigma_1} \Sigma_a, \delta_3, q_0, F_1, \nu_3)$, where $K_3 = K_1 \cup \bigcup_{a \text{ in } \Sigma_1} (K_1 \times K_a)$, $\nu_3 = (\alpha_1, \ldots, \alpha_k, \beta_1, \ldots, \beta_\ell)$, and δ_3 is defined as follows (for each q in K_1, a in Σ_1, and $(\gamma_1, \ldots, \gamma_k)$ in G_{D_1}):

(1) Let $(q',(u_1, \ldots, u_k, 1_{\beta_1}, \ldots, 1_{\beta_\ell}))$ be in $\delta_3(q,\epsilon,(\gamma_1, \ldots, \gamma_k, \epsilon, \ldots, \epsilon))$ if $(q',(u_1, \ldots, u_k))$ is in $\delta_1(q,\epsilon,(\gamma_1, \ldots, \gamma_k))$.

(2) Let $((q,q_a), (1(\alpha_1,\gamma_1), \ldots, 1(\alpha_k,\gamma_k), 1_{\beta_1}, \ldots, 1_{\beta_\ell}))$ be in $\delta_3(q,\epsilon,(\gamma_1, \ldots, \gamma_k, \epsilon, \ldots, \epsilon))$.

(3) Let $((q,p'), (1(\alpha_1,\gamma_1), \ldots, 1(\alpha_k,\gamma_k), u_1, \ldots, u_\ell))$ be in $\delta_3((q,p),b,(\gamma_1, \ldots, \gamma_k,\gamma_1', \ldots,\gamma_\ell'))$ if $(p',(u_1, \ldots, u_\ell))$ is in $\delta_a(p,b,(\gamma_1',\ldots,\gamma_\ell'))$, where p is in K_a, b in Σ_a, and $(\gamma_1', \ldots, \gamma_\ell')$ in G_{D_a}.

(4) Let $(q',(u_1, \ldots, u_k, 1_{\beta_1}, \ldots, 1_{\beta_\ell}))$ be in $\delta_3((q,p),\epsilon,(\gamma_1,\ldots,\gamma_k,\epsilon,\ldots,\epsilon))$ if p is in F_a and $(q',(u_1, \ldots,u_k))$ is in $\delta_1(q,a,(\gamma_1, \ldots, \gamma_k))$.

Clearly $L(D_3) = \tau(L(D_1))$. Since D_1 and the D_a are nested, so is D_3. If D_1 and the D_a are quasi-realtime, and τ is ϵ-free, then D_3 is quasi-realtime. (For if D_1 has at most k_1 ϵ-moves and each D_a at most k_a, then D_3 has at most $2\max\{k_a/a\} + k_1 + 2$ consecutive ϵ-moves).

The reverse inclusions of Lemma 4.2 are also true. That is, we have

<u>Lemma 4.3</u>. Under the hypotheses of Lemma 4.2,

$$\mathcal{L}^t(\mathcal{D}_3^N) \subseteq \mathcal{L}^t(\mathcal{D}_1^N) \ \sigma \ \mathcal{L}^t(\mathcal{D}_2^N)$$

and $\mathcal{L}(\mathcal{D}_3^N) \subseteq \mathcal{L}(\mathcal{D}_1^N) \ \sigma \ \mathcal{L}(\mathcal{D}_2^N)$.

The proof of Lemma 4.3 is quite involved and is not especially enlightening. As such, it is given in the appendix.

<u>Lemma 4.4</u>. Let $(\Omega_1,\mathcal{D}_1)$, ..., $(\Omega_n,\mathcal{D}_n)$ be multitape AFA, with $n\geq 2$. Then

$$\mathcal{L}^t((\mathcal{D}_1 \wedge \ldots \wedge \mathcal{D}_n)^N) = \mathcal{L}^t(\mathcal{D}_1^N) \ \sigma \ \mathcal{L}^t(\mathcal{D}_2^N) \ \sigma \ \ldots \ \sigma \ \mathcal{L}^t(\mathcal{D}_n^N)$$

and
$$\begin{aligned}\mathcal{L}((\mathcal{D}_1 \wedge \ldots \wedge \mathcal{D}_n)^N) &= \mathcal{L}(\mathcal{D}_1^N) \ \hat{\sigma} \ \mathcal{L}(\mathcal{D}_2^N) \ \hat{\sigma} \ \ldots \ \hat{\sigma} \ \mathcal{L}(\mathcal{D}_n^N) \\ &= \mathcal{L}(\mathcal{D}_1^N) \ \sigma \ \mathcal{L}(\mathcal{D}_2^N) \ \sigma \ \ldots \ \sigma \ \mathcal{L}(\mathcal{D}_n^N) \\ &= \mathcal{L}^t(\mathcal{D}_1^N) \ \hat{\sigma} \ (\mathcal{L}(\mathcal{D}_2^N) \ \sigma \ \ldots \ \sigma \ \mathcal{L}(\mathcal{D}_n^N)).\end{aligned}$$

<u>Proof</u>. Suppose $n=2$. By Lemmas 4.2 and 4.3,

$$\mathcal{L}^t((\mathcal{D}_1 \wedge \mathcal{D}_2)^N) = \mathcal{L}^t(\mathcal{D}_1^N) \ \sigma \ \mathcal{L}^t(\mathcal{D}_2^N)$$

and $\mathcal{L}(\mathcal{D}_1^N) \ \hat{\sigma} \ \mathcal{L}(\mathcal{D}_2^N) \subseteq \mathcal{L}((\mathcal{D}_1 \wedge \mathcal{D}_2)^N) \subseteq \mathcal{L}(\mathcal{D}_1^N) \ \sigma \ \mathcal{L}(\mathcal{D}_2^N)$.

Since $\mathcal{L}(\mathcal{D}_1^N) \ \sigma \ \mathcal{L}(\mathcal{D}_2^N) \subseteq \mathcal{L}(\mathcal{D}_1^N) \ \hat{\sigma} \ \mathcal{L}(\mathcal{D}_2^N)$,

$$\begin{aligned}\mathcal{L}(\mathcal{D}_1^N) \ \sigma \ \mathcal{L}(\mathcal{D}_2^N) = \mathcal{L}(\mathcal{D}_1^N) \ \hat{\sigma} \ \mathcal{L}(\mathcal{D}_2^N) &= \mathcal{L}((\mathcal{D}_1 \wedge \mathcal{D}_2)^N) \\ &= \hat{\mathcal{H}}(\mathcal{L}^t(\mathcal{D}_1^N)) \ \hat{\sigma} \ \mathcal{L}(\mathcal{D}_2^N) \\ &= \mathcal{L}^t(\mathcal{D}_1^N) \ \hat{\sigma} \ \mathcal{L}(\mathcal{D}_2^N),\end{aligned}$$ by Remark 5 following the definition of substitution.

Continuing by induction, suppose the result is true for $n-1\geq 2$. Consider n. Then

$$\mathcal{L}^t((\mathfrak{s}_1\wedge \ldots \wedge \mathfrak{s}_n)^N) = \mathcal{L}^t((\mathfrak{s}_1\wedge \ldots \wedge \mathfrak{s}_{n-1})^N)\ \sigma\ \mathcal{L}^t(\mathfrak{s}_n^N), \text{ by induction,}$$
$$= \mathcal{L}^t(\mathfrak{s}_1^N)\ \sigma \ldots \sigma\ \mathcal{L}^t(\mathfrak{s}_{n-1}^N)\ \sigma\ \mathcal{L}^t(\mathfrak{s}_n^N), \text{ by induction,}$$

and

$$\mathcal{L}((\mathfrak{s}_1 \wedge \ldots \wedge \mathfrak{s}_n)^N) = \mathcal{L}((\mathfrak{s}_1\wedge \ldots \wedge \mathfrak{s}_{n-1})^N)\ \sigma\ \mathcal{L}(\mathfrak{s}_n^N), \text{ by induction,}$$
$$= (\mathcal{L}(\mathfrak{s}_1^N)\ \sigma \ldots \sigma\ \mathcal{L}(\mathfrak{s}_{n-1}^N))\ \sigma\ \mathcal{L}(\mathfrak{s}_n^N), \text{ by induction,}$$
$$= \mathcal{L}(\mathfrak{s}_1^N)\ \hat{\sigma} \ldots \hat{\sigma}\ \mathcal{L}(\mathfrak{s}_n^N), \text{ by Remark 5,}$$
$$= \hat{H}(\mathcal{L}^t(\mathfrak{s}_1^N))\ \hat{\sigma}\ (\mathcal{L}(\mathfrak{s}_2^N)\ \hat{\sigma} \ldots \hat{\sigma}\ \mathcal{L}(\mathfrak{s}_n^N)), \text{ by Remark 5,}$$
$$= \hat{H}(\mathcal{L}^t(\mathfrak{s}_1^N))\ \hat{\sigma}\ (\mathcal{L}(\mathfrak{s}_2^N)\ \sigma \ldots \sigma\ \mathcal{L}(\mathfrak{s}_n^N)), \text{ by induction,}$$
$$= \mathcal{L}^t(\mathfrak{s}_1^N)\ \hat{\sigma}\ (\mathcal{L}(\mathfrak{s}_2^N)\ \sigma \ldots \sigma\ \mathcal{L}(\mathfrak{s}_n^N)), \text{ by Remark 5.}$$

Since $\mathfrak{s}^N = \mathfrak{s}$ for a single-tape AFA, we immediately get

<u>Theorem 4.2</u>. Let $(\Omega_1,\mathfrak{s}_1), \ldots, (\Omega_n,\mathfrak{s}_n)$ be single-tape AFA, with $n\geq 2$. Then

$$\mathcal{L}^t((\mathfrak{s}_1\wedge \ldots \wedge \mathfrak{s}_n)^N) = \mathcal{L}^t(\mathfrak{s}_1)\ \sigma\ \mathcal{L}^t(\mathfrak{s}_2)\ \sigma \ldots \sigma\ \mathcal{L}^t(\mathfrak{s}_n)$$

and

$$\mathcal{L}((\mathfrak{s}_1\wedge \ldots \wedge \mathfrak{s}_n)^N) = \mathcal{L}(\mathfrak{s}_1)\ \hat{\sigma}\ \mathcal{L}(\mathfrak{s}_2)\ \hat{\sigma} \ldots \hat{\sigma}\ \mathcal{L}(\mathfrak{s}_n)$$
$$= \mathcal{L}(\mathfrak{s}_1)\ \sigma\ \mathcal{L}(\mathfrak{s}_2)\ \sigma \ldots \sigma\ \mathcal{L}(\mathfrak{s}_n)$$
$$= \mathcal{L}^t(\mathfrak{s}_1)\ \hat{\sigma}\ (\mathcal{L}(\mathfrak{s}_2)\ \sigma \ldots \sigma\ \mathcal{L}(\mathfrak{s}_n)).$$

From Theorem 4.2, we derive

<u>Corollary 1</u>. If $\mathcal{L}_1, \ldots, \mathcal{L}_n$, $n\geq 2$, are AFL, then so is $\mathcal{L}_1\sigma\ \mathcal{L}_2\sigma \ldots \sigma\ \mathcal{L}_n$.

<u>Proof</u>. By induction, it suffices to show the result for n=2.

Consider n=2. There exist single-tape AFA $\mathcal{D}_1$ and $\mathcal{D}_2$ such that $\mathcal{L}^t(\mathcal{D}_i) = \mathcal{L}_i \cup \{L \cup \{\epsilon\}/L \text{ in } \mathcal{L}_i\}$ for each i [5]. By Theorem 4.2,

$$\mathcal{L}^t(\mathcal{D}_1) \;\sigma\; \mathcal{L}^t(\mathcal{D}_2) = \mathcal{L}^t((\mathcal{D}_1 \wedge \mathcal{D}_2)^N),$$

so that $\mathcal{L}^t(\mathcal{D}_1) \;\sigma\; \mathcal{L}^t(\mathcal{D}_2)$ is an AFL. By definition of the operation σ (ϵ-free substitution),

$$\mathcal{L}^t(\mathcal{D}_1) \;\sigma\; \mathcal{L}^t(\mathcal{D}_2) = \mathcal{L}^t(\mathcal{D}_1) \;\sigma\; \mathcal{L}_2$$
$$= (\mathcal{L}_1 \sigma\, \mathcal{L}_2) \cup (\{L \cup \{\epsilon\}/L \text{ in } \mathcal{L}_1\} \;\sigma\; \mathcal{L}_2).$$

If $\mathcal{L}_1$ contains ϵ, then $\mathcal{L}_1 \sigma\, \mathcal{L}_2 = \mathcal{L}^t(\mathcal{D}_1) \;\sigma\; \mathcal{L}^t(\mathcal{D}_2)$. If $\mathcal{L}_1$ does not contain $\{\epsilon\}$, then $\mathcal{L}_1 \sigma\, \mathcal{L}_2 = \{L - \{\epsilon\}/L \text{ in } \mathcal{L}^t(\mathcal{D}_1) \;\sigma\; \mathcal{L}^t(\mathcal{D}_2)\}$.

In either case, $\mathcal{L}_1 \sigma\, \mathcal{L}_2$ is an AFL.

<u>Corollary 2</u>. If $\mathcal{L}_1, \ldots, \mathcal{L}_n$, $n \geq 2$, are AFL, then so is $\mathcal{L}_1 \hat{\sigma}\, \mathcal{L}_2 \hat{\sigma} \ldots \hat{\sigma}\, \mathcal{L}_n$.

<u>Proof</u>. Again it suffices to consider the case n=2. If $\mathcal{L}_2$ does not contain $\{\epsilon\}$, then $\mathcal{L}_1 \hat{\sigma}\, \mathcal{L}_2 = \mathcal{L}_1 \sigma\, \mathcal{L}_2$. If $\mathcal{L}_2$ contains $\{\epsilon\}$, then $\mathcal{L}_1 \hat{\sigma}\, \mathcal{L}_2 = \hat{H}(\mathcal{L}_1) \;\sigma\; \mathcal{L}_2$ by Remark 5. In either case, $\mathcal{L}_1 \hat{\sigma}\, \mathcal{L}_2$ is an AFL by Corollary 1.

<u>Corollary 3</u>. If $\mathcal{L}_1$ is an AFL and $\mathcal{L}_2$ is a full AFL, then $\mathcal{L}_1 \hat{\sigma}\, \mathcal{L}_2$ is a full AFL.

<u>Proof</u>. Let $\mathcal{D}_1$ and $\mathcal{D}_2$ be AFA such that $\mathcal{L}^t(\mathcal{D}_1) = \mathcal{L}_1 \cup \{L \cup \{\epsilon\}/L \text{ in } \mathcal{L}_1\}$ and $\mathcal{L}(\mathcal{D}_2) = \mathcal{L}_2$. By Theorem 4.2, $\mathcal{L}^t(\mathcal{D}_1) \;\hat{\sigma}\; \mathcal{L}(\mathcal{D}_2) = \mathcal{L}((\mathcal{D}_1 \wedge \mathcal{D}_2)^N)$, so that $\mathcal{L}^t(\mathcal{D}_1) \;\hat{\sigma}\; \mathcal{L}(\mathcal{D}_2)$ is a full AFL. Now

$$\mathcal{L}^t(\mathcal{D}_1) \;\hat{\sigma}\; \mathcal{L}(\mathcal{D}_2) = (\mathcal{L}_1 \hat{\sigma}\, \mathcal{L}_2) \cup \{L \cup \{\epsilon\}/L \text{ in } \mathcal{L}_1 \hat{\sigma}\, \mathcal{L}_2\}.$$

By Corollary 2, $\mathcal{L}_1 \hat{\sigma}\, \mathcal{L}_2$ is an AFL. Since $\{\epsilon\}$ is in $\mathcal{L}_1 \hat{\sigma}\, \mathcal{L}_2$ ($\{\epsilon\} = \tau(\{a\})$ for the substitution $\tau(a) = \{\epsilon\}$), $\mathcal{L}_1 \hat{\sigma}\, \mathcal{L}_2 = \mathcal{L}^t(\mathcal{D}_1)\ \hat{\sigma}\ \mathcal{L}(\mathcal{D}_2)$.

<u>Remark.</u> Corollaries 2 and 3 were proved in [13] by different methods. By Remark 1 of Section 4, Corollary 2 implies Corollary 1.

<u>Corollary 4.</u> Let $(\Omega,\mathcal{D})$ be a multitape AFA, with $\Omega = (K,\Sigma,\mathcal{A},<,\mu)$. Then

$$\mathcal{L}^t(\mathcal{D}^N) = \bigcup_{\substack{n\geq 2 \\ \alpha_i \text{ in } \mathcal{A} \\ \alpha_1 < \cdots < \alpha_n}} \mathcal{L}^t(\mathcal{D}_{\alpha_1})\ \sigma \cdots \sigma\ \mathcal{L}^t(\mathcal{D}_{\alpha_n})$$

and
$$\mathcal{L}(\mathcal{D}^N) = \bigcup_{\substack{n\geq 2 \\ \alpha_i \text{ in } \mathcal{A} \\ \alpha_1 < \cdots < \alpha_n}} \mathcal{L}(\mathcal{D}_{\alpha_1})\ \sigma \cdots \sigma\ \mathcal{L}(\mathcal{D}_{\alpha_n}).$$

If $\mathcal{A}$ is finite, say $\mathcal{A} = \{\alpha_1, \ldots, \alpha_n\}$, then

$$\mathcal{L}^t(\mathcal{D}^N) = \mathcal{L}^t(\mathcal{D}_{\alpha_1})\ \sigma \cdots \sigma\ \mathcal{L}^t(\mathcal{D}_{\alpha_n})$$

and
$$\begin{aligned}\mathcal{L}(\mathcal{D}^N) &= \mathcal{L}(\mathcal{D}_{\alpha_1})\ \sigma \cdots \sigma\ \mathcal{L}(\mathcal{D}_{\alpha_n}) \\ &= \mathcal{L}(\mathcal{D}_{\alpha_1})\ \hat{\sigma} \cdots \hat{\sigma}\ \mathcal{L}(\mathcal{D}_{\alpha_n}).\end{aligned}$$

We now turn to the representation of the "substitution closure" of $\mathcal{L}^t(\mathcal{D})$ and $\mathcal{L}(\mathcal{D})$.

<u>Notation.</u> For each family of languages $\mathcal{L}$, let

(a) $\mathcal{S}(\mathcal{L})$ be the smallest AFL containing $\mathcal{L}$ and closed under ϵ-free substitution.

(b) $\hat{\mathcal{S}}(\mathcal{L})$ be the smallest full AFL containing $\mathcal{L}$ and closed under substitution.

(c) $\sigma(\mathcal{L}) = \bigcup_{n\geq 1} \sigma_n(\mathcal{L})$, where $\sigma_1(\mathcal{L}) = \mathcal{L} \sigma \mathcal{L}$ and $\sigma_{i+1}(\mathcal{L}) = \sigma_i(\mathcal{L}) \sigma \mathcal{L}$ for each $i\geq 1$.

(d) $\hat{\sigma}(\mathcal{L}) = \bigcup_{n\geq 1} \hat{\sigma}_n(\mathcal{L})$, where $\hat{\sigma}_1(\mathcal{L}) = \mathcal{L} \hat{\sigma} \mathcal{L}$ and $\hat{\sigma}_{i+1}(\mathcal{L}) = \hat{\sigma}_i(\mathcal{L}) \hat{\sigma} \mathcal{L}$ for each $i\geq 1$.

Thus $\mathcal{S}(\mathcal{L})(\hat{\mathcal{S}}(\mathcal{L}))$ is the ϵ-free substitution (substitution) closure AFL generated by $\mathcal{L}$. $\sigma_n(\mathcal{L})$ $(\hat{\sigma}_n(\mathcal{L}))$ is the "n-th level of ϵ-free substitution (arbitrary substitution) of $\mathcal{L}$ into itself."

If $\mathcal{L}$ is an AFL, then from Remark 2 following the definition of substitution, $\sigma_n(\mathcal{L}) = \mathcal{L} \sigma \ldots \sigma \mathcal{L}$ (n occurrences of σ) and $\hat{\sigma}_n(\mathcal{L}) = \mathcal{L} \hat{\sigma} \ldots \hat{\sigma} \mathcal{L}$ (n occurrences of $\hat{\sigma}$).

We now present the result relating substitution closure and nested multitape AFA.

Theorem 4.3. Let $\mathcal{D}$ be a single-tape AFA. Then

$$\mathcal{L}^t((\wedge\mathcal{D})^N) = \sigma(\mathcal{L}^t(\mathcal{D})) = \mathcal{S}(\mathcal{L}^t(\mathcal{D}))$$

and
$$\mathcal{L}((\wedge\mathcal{D})^N) = \hat{\sigma}(\mathcal{L}(\mathcal{D})) = \hat{\mathcal{S}}(\mathcal{L}(\mathcal{D}))$$
$$= \sigma(\mathcal{L}(\mathcal{D})) = \mathcal{S}(\mathcal{L}(\mathcal{D})).$$

Proof. For each $i\geq 1$, let $\mathcal{D}_i = \mathcal{D}$. Then

$$\mathcal{L}^t((\wedge\mathcal{D})^N) = \bigcup_{n\geq 1} \mathcal{L}^t((\mathcal{D}_1 \wedge \ldots \wedge \mathcal{D}_n)^N)$$

$$= \bigcup_{n\geq 1} \mathcal{L}^t(\mathcal{D}_1)\ \sigma\ \ldots\ \sigma\ \mathcal{L}^t(\mathcal{D}_n)),\ \text{by Theorem 4.2,}$$

$$= \sigma(\mathcal{L}^t(\mathcal{D})),$$

and $\mathcal{L}((\wedge\mathcal{D})^N) = \bigcup_{n\geq 1} \mathcal{L}((\mathcal{D}_1\wedge \ldots \wedge \mathcal{D}_n)^N)$

$$= \bigcup_{n\geq 1} (\mathcal{L}(\mathcal{D}_1)\ \hat{\sigma}\ \ldots\ \hat{\sigma}\ \mathcal{L}(\mathcal{D}_n)),\ \text{by Theorem 4.2,}$$

$$= \hat{\sigma}(\mathcal{L}(\mathcal{D}))$$

$$= \bigcup_{n\geq 1} (\mathcal{L}(\mathcal{D}_1)\ \sigma\ \ldots\ \sigma\ \mathcal{L}(\mathcal{D}_n)),\ \text{by Theorem 4.2,}$$

$$= \sigma(\mathcal{L}(\mathcal{D})).$$

Since $\sigma(\mathcal{L}^t(\mathcal{D})) = \mathcal{L}^t((\wedge\mathcal{D})^N)$ is an AFL containing $\mathcal{L}^t(\mathcal{D})$ and closed under ϵ-free substitution, $\mathcal{S}(\mathcal{L}^t(\mathcal{D})) \subseteq \sigma(\mathcal{L}^t(\mathcal{D}))$. Obviously $\sigma(\mathcal{L}^t(\mathcal{D})) \subseteq \mathcal{S}(\mathcal{L}^t(\mathcal{D}))$, so that $\sigma(\mathcal{L}^t(\mathcal{D})) = \mathcal{S}(\mathcal{L}^t(\mathcal{D}))$. By similar reasoning, $\hat{\sigma}(\mathcal{L}(\mathcal{D})) = \hat{\mathcal{S}}(\mathcal{L}(\mathcal{D}))$ and $\sigma(\mathcal{L}(\mathcal{D})) = \mathcal{S}(\mathcal{L}(\mathcal{D}))$, completing the proof.

Corollary 1. For each AFL $\mathcal{L}$, $\sigma(\mathcal{L}) = \mathcal{S}(\mathcal{L})$, and $\sigma(\mathcal{L})$ and $\hat{\sigma}(\mathcal{L})$ are AFL.

Proof. By Corollary 1 of Theorem 4.2, $\sigma_n(\mathcal{L})$ is an AFL for each $n\geq 1$. Since $\sigma_n(\mathcal{L}) \subseteq \sigma_{n+1}(\mathcal{L})$ for each $n\geq 1$, $\sigma(\mathcal{L}) = \bigcup_{n\geq 1} \sigma_n(\mathcal{L})$ is an AFL. Similarly, using Corollary 2 of Theorem 4.2, $\hat{\sigma}(\mathcal{L})$ is an AFL.

Clearly $\sigma_n(\mathcal{L}) \subseteq \mathcal{S}(\mathcal{L})$ for each $n\geq 1$. Thus $\sigma(\mathcal{L}) \subseteq \mathcal{S}(\mathcal{L})$. Since $\sigma(\mathcal{L})$ is closed under ϵ-free substitution, $\mathcal{S}(\mathcal{L}) \subseteq \sigma(\mathcal{L})$, whence equality.

Remark. (1) $\hat{\sigma}(\mathcal{L})$ need not be a full AFL. For let $\mathcal{L}$ be the family of context-sensitive languages. Then $\mathcal{L}$ is an AFL and $\hat{\sigma}(\mathcal{L}) = \mathcal{L}$, but $\mathcal{L}$ is not a full AFL. The next corollary shows that $\hat{\sigma}(\mathcal{L})$ is a full AFL if $\mathcal{L}$ is a full AFL.

(2) It was shown in [13] that $\hat{\sigma}(\mathcal{L})$ is an AFL if $\mathcal{L}$ is an AFL.

Corollary 2. If $\mathcal{L}$ is a full AFL, then $\hat{\sigma}(\mathcal{L}) = \sigma(\mathcal{L}) = \hat{\mathcal{S}}(\mathcal{L}) = \mathcal{S}(\mathcal{L})$ and $\hat{\sigma}(\mathcal{L})$ is a full AFL.

Proof. By Remark 4 following the definition of substitution and by Corollary 3 of Theorem 4.2, $\hat{\sigma}_n(\mathcal{L}) = \sigma_n(\mathcal{L})$ and $\hat{\sigma}_n(\mathcal{L})$ is a full AFL for each $n \geq 1$. Thus $\sigma(\mathcal{L}) = \bigcup_{n\geq 1} \sigma_n(\mathcal{L}) = \bigcup_{n\geq 1} \hat{\sigma}_n(\mathcal{L}) = \hat{\sigma}(\mathcal{L})$. Since each $\hat{\sigma}_n(\mathcal{L}) \subseteq \hat{\sigma}_{n+1}(\mathcal{L})$ for each $n \geq 1$, $\hat{\sigma}(\mathcal{L})$ is a full AFL. Clearly $\hat{\sigma}(\mathcal{L}) \subseteq \hat{\mathcal{S}}(\mathcal{L})$. Since $\hat{\sigma}(\mathcal{L})$ is closed under substitution $\hat{\mathcal{S}}(\mathcal{L}) \subseteq \hat{\sigma}(\mathcal{L})$. Hence $\hat{\sigma}(\mathcal{L}) = \hat{\mathcal{S}}(\mathcal{L})$. Similarly $\sigma(\mathcal{L}) = \mathcal{S}(\mathcal{L})$.

Remark. It was shown in [13] that $\hat{\sigma}(\mathcal{L})$ is a full AFL and $\hat{\sigma}(\mathcal{L}) = \hat{\mathcal{S}}(\mathcal{L})$, if $\mathcal{L}$ is a full AFL.

Corollary 3. If $\mathcal{L}$ is an ϵ-free AFL, then $\hat{\sigma}(\mathcal{L}) = \mathcal{S}(\mathcal{L})$. If $\mathcal{L}$ is an AFL containing $\{\epsilon\}$, then

$$\hat{\sigma}(\mathcal{L}) = \hat{\sigma}\,\hat{\mathcal{H}}(\mathcal{L}) = \hat{\mathcal{S}}(\mathcal{L}) = \mathcal{S}\hat{\mathcal{H}}(\mathcal{L})$$

and $\hat{\sigma}(\mathcal{L})$ is a full AFL.

Proof. Let $\mathcal{L}$ be an AFL. If $\mathcal{L}$ is ϵ-free, then $\sigma_n(\mathcal{L}) = \hat{\sigma}_n(\mathcal{L})$ for each n, so that $\hat{\sigma}(\mathcal{L}) = \sigma(\mathcal{L})$. By Corollary 1, $\sigma(\mathcal{L}) = \mathcal{S}(\mathcal{L})$.

Suppose $\mathcal{L}$ contains $\{\epsilon\}$. Clearly $\hat{\sigma}(\mathcal{L}) \subseteq \hat{\sigma}\,\hat{\mathcal{H}}(\mathcal{L}) \subseteq \hat{\mathcal{S}}(\mathcal{L}) \subseteq \hat{\mathcal{S}}\hat{\mathcal{H}}(\mathcal{L})$. By Corollary 2, $\hat{\sigma}\,\hat{\mathcal{H}}(\mathcal{L}) = \hat{\mathcal{S}}\hat{\mathcal{H}}(\mathcal{L}) = \mathcal{S}\hat{\mathcal{H}}(\mathcal{L})$ and $\hat{\sigma}\,\hat{\mathcal{H}}(\mathcal{L})$ is a full AFL. Thus

$$\hat{\sigma}(\mathcal{L}) \subseteq \hat{\sigma}\,\hat{\mathcal{H}}(\mathcal{L}) = \hat{\mathcal{S}}(\mathcal{L}) = \mathcal{S}\hat{\mathcal{H}}(\mathcal{L})$$

and $\hat{\sigma}\,\hat{\mathcal{H}}(\mathcal{L})$ is a full AFL. It thus suffices to show that $\hat{\sigma}\,\hat{\mathcal{H}}(\mathcal{L}) \subseteq \hat{\sigma}(\mathcal{L})$. Now

$$\hat{\mathcal{H}}(\mathcal{L}) \subseteq \hat{\mathcal{H}}(\mathcal{L})\,\hat{\sigma}\,\mathcal{L}$$

$$= \mathcal{L}\,\hat{\sigma}\,\mathcal{L},$$ since $\mathcal{L}$ contains $\{\epsilon\}$ (by Remark 5 following the definition of substitution).

Let $\mathcal{L}_i = \mathcal{L}$ for each $i \geq 1$. Then for each $n \geq 1$,

$$\hat{\mathcal{H}}(\mathcal{L}_1) \hat{\sigma} \ldots \hat{\sigma} \hat{\mathcal{H}}(\mathcal{L}_{n+1}) \subseteq (\mathcal{L}_1 \hat{\sigma} \mathcal{L}_1) \hat{\sigma} \ldots \hat{\sigma} (\mathcal{L}_{n+1} \hat{\sigma} \mathcal{L}_{n+1})$$
$$\subseteq \hat{\sigma}(\mathcal{L}).$$

Hence $\hat{\sigma}\, \hat{\mathcal{H}}(\mathcal{L}) \subseteq \hat{\sigma}(\mathcal{L})$.

Remark. It was shown in [13] that $\hat{\sigma}(\mathcal{L}) = \mathcal{S}(\mathcal{L})$ if $\mathcal{L}$ is an ϵ-free AFL.

Examples. (1) Counters. In [15] the notion of a 1-counter acceptor is generalized to that of a pda whose storage configurations are limited to the bounded regular sets $Z_0 A_1^* \ldots A_n^*$. This device is easily seen to be equivalent to a nested acceptor with n counters. Given $n \geq 0$, the family of languages defined by $\mathcal{L}((\mathcal{D}_1 \wedge \ldots \wedge \mathcal{D}_n)^N)$, each $\mathcal{D}_i$ a 1-counter AFA, is denoted by $\mathcal{F}_{n,\omega}$ and $\mathcal{L}((\wedge \mathcal{D}_1)^N)$ by $\mathcal{F}_{\infty,\omega}$ [15]. It is shown that $\mathcal{F}_{n,\omega}$ is properly contained in $\mathcal{F}_{n+1,\omega}$ for each $n \geq 0$. By Theorem 4.2, $\mathcal{F}_{n+m,\omega} = \mathcal{F}_{n,\omega} \sigma \mathcal{F}_{m,\omega} = \mathcal{F}_{n,\omega} \hat{\sigma} \mathcal{F}_{m,\omega}$ for all n, $m \geq 1$. By Theorem 4.3, $\mathcal{F}_{\infty,\omega} = \hat{\mathcal{S}}(\mathcal{F}_{1,\omega})$.

(2) Linear context-free languages. A family that has recently been studied from three different viewpoints is $\mathcal{L} = \hat{\mathcal{S}}(\mathcal{H}_\ell)$, where $\mathcal{H}_\ell$ is the family of linear context-free languages. Note that $\mathcal{H}_\ell$ is not an AFL since it is not closed under concatenation. $\mathcal{L}$ is called the "standard matching choice languages" [21], the "quasi-rational languages" [20], and the "derivation-bounded languages" [12]. Yntema and Nivat, independently, proved that $\mathcal{L}$ is properly contained in the context-free languages. Theorem 4.3 allows us to give a fairly simple acceptor realization for $\mathcal{L}$.

Let $\mathcal{F}_{\omega,1} = \hat{\mathcal{F}}(\{wcw^R/w \text{ in } \{a,b\}^*\})$. $\mathcal{F}_{\omega,1}$ is the smallest full AFL

containing $\mathcal{H}_\ell$ [2, 15, 20]. Thus $\mathcal{L} = \hat{\mathcal{S}}(\mathcal{F}_{\omega,1})$. Let K be an infinite denumerable set. Let ξ be a new symbol and Γ an infinite set containing ξ. Let $I = (\Gamma-\{\xi\})^*$. Let $(\Omega,\mathcal{D})$, with $\Omega = (K,\Sigma,\Gamma,I,f,g)$, be the (single-tape) AFA where f and g are defined as follows (for all w in $(\Gamma-\{\xi\})^*$, y in $(\Gamma-\{\xi\})^+$, and Y,Z in $\Gamma-\{\xi\}$):

(1) $g(\epsilon) = \{\epsilon\}$ and $g(wZ) = g(\xi wZ) = Z$.

(2) $f(wZ,y) = wy$, $f(\epsilon,w) = w$, $f(\xi wZ,Y) = \xi wY$, and $f(\xi Z,\epsilon) = f(Z,\epsilon) = \epsilon$.

(3) $f(wZ,\epsilon) = f(\xi wZ,\epsilon) = \xi w$ for $w\neq\epsilon$.

Clearly $(\Omega,\mathcal{D})$ is a one-turn bounded pda AFA, that is, each D in $\mathcal{D}$ can make at most one turn (i.e., the length of the storage configuration changes at most once from increasing to decreasing) before returning to the storage configuration ϵ [10, 15]. It is shown in [15] that $\mathcal{F}_{\omega,1} = \mathcal{L}(\mathcal{D})$. It can also be shown, although not done here, that $\mathcal{F}_{\omega,1} = \mathcal{L}^t(\mathcal{D})$. Let $\mathcal{L}_1 = \mathcal{F}_{\omega,1}$ and $\mathcal{L}_{n+1} = \mathcal{L}_n \hat{\sigma}\, \mathcal{L}_1$ for each $n\geq 1$. (Each $\mathcal{L}_n$ is the family of quasi-rational languages of order n [20].) By Theorem 4.3, $\mathcal{L} = \mathcal{L}((\wedge\mathcal{D})^N) = \bigcup_n \mathcal{L}_n = \sigma(\mathcal{L}(\mathcal{D})) = \sigma(\mathcal{L}^t(\mathcal{D})) = \mathcal{L}^t((\wedge\mathcal{D})^N)$. By Theorem 4.2, each $\mathcal{L}_n$ is a full AFL. A result of Greibach [17] asserts that if $\mathcal{L}_1$ is not closed under substitution, then each $\mathcal{L}_n$ is properly contained in $\mathcal{L}_{n+1}$ and $\mathcal{S}(\mathcal{L}_1) = \mathcal{S}(\mathcal{H}_\ell) = \mathcal{L}$ is properly contained in the family of context-free languages.

(3) <u>One-way stack languages</u>. Let $(\Omega_S,\mathcal{D}_S)$ be the one-way stack AFA and $\mathcal{L}_S = \mathcal{L}(\mathcal{D}_S)$ [6, 5]. Thus $\mathcal{L}_S$ is the one-way stack languages. It was shown [16] that $\mathcal{L}_S$ is not closed under substitution. Hence $\mathcal{L}_S \neq \mathcal{L}_S \hat{\sigma}\, \mathcal{L}_S = \mathcal{L}((\mathcal{D}_S \wedge \mathcal{D}_S)^N)$ and $\mathcal{L}(\mathcal{D}_S) \neq \mathcal{L}((\wedge\mathcal{D}_S)^N)$.

The nested stack acceptors (nsa) of [1] are far more general devices than the acceptors in $(\wedge\mathcal{A}_S)^N$. Members of $(\wedge\mathcal{A}_S)^N$ are essentially nsa which are

(a) nested (as defined in Section 4--there is at most one nest of stacks at any time.

(b) finitely nested--for each D there is an n such that no more than n stacks are active at any time.

Both (a) and (b) restrict the power of nsa. Specifically, it is shown in [17] that the nsa languages properly include $\hat{\sigma}(\mathcal{L}_S)$.

APPENDIX

We consider here the proof of Lemma 4.3. In the process we shall need some notation and ideas pertinent only to this appendix. In addition, we shall need two preliminary lemmas.

We henceforth assume that $\mathcal{D}_1$ and $\mathcal{D}_2$ are given multitape AFA and that $\mathcal{D}_3 = \mathcal{D}_1 \wedge \mathcal{D}_2$.

Let $D = (K_1, \Sigma_1, \delta, q_0, F, \nu)$ be an acceptor in $\mathcal{D}_3^N$ and let $\nu = (\alpha_1, \ldots, \alpha_k, \beta_1, \ldots, \beta_\ell)$, with $k, \ell \geq 1$, each α_i in $\mathcal{a}_1$ and each β_j in $\mathcal{a}_2$. Let $G_D' = \{(\gamma_1, \ldots, \gamma_k)/(\gamma_1, \ldots, \gamma_k, \gamma_1', \ldots, \gamma_\ell')$ in G_D for some $\gamma_1', \ldots, \gamma_\ell'\}$. For each G, $\emptyset \neq G \subseteq G_D'$, write

$$(q, a, w, (y_1, \ldots, y_k, y_1', \ldots, y_\ell')) \vdash_{\overline{G}} (q', w, (y_1, \ldots, y_k, \bar{y}_1', \ldots, \bar{y}_\ell'))$$

if there exist $\gamma_1, \ldots, \gamma_k, \gamma_1', \ldots, \gamma_\ell', u_1, \ldots, u_k, u_1', \ldots, u_\ell'$ such that

(1) $(q', (u_1, \ldots, u_k, u_1', \ldots, u_\ell'))$ is in $\delta(q, a, (\gamma_1, \ldots, \gamma_k, \gamma_1', \ldots, \gamma_\ell'))$;

(2) $\bar{y}_j' = f_{\beta_j}(y_j', u_j')$ for each j, $1 \leq j \leq \ell$;

(3) $(\gamma_1, \ldots, \gamma_k)$ is in G;

(4) u_i is in $\psi_{\alpha_i}(\gamma_i)$ and γ_i is in $g_{\alpha_i}(y_i)$ for each i, and γ_j' is in $g_{\beta_j}(y_j')$ for each j;

and (5) $y_{j_0}' \neq \epsilon$ or $\bar{y}_{j_0}' \neq \epsilon$ for some j_0.

For each $i \geq 0$ let $\overset{i}{\vdash}$ be the relation on configurations defined by induction

as follows: $C \vdash_G^{0} C$ for each C and $C \vdash_G^{n+1} C'$ if there exists C'' such that $C \vdash_G^{n} C''$ and $C'' \vdash_G C'$. Let $\vdash_G^{*}$ be the transitive, reflexive extension of $\vdash_G$. Let $\vdash_{,}$ be the relation defined by

$$(q, aw, (y_1, \ldots, y_k, \epsilon, \ldots, \epsilon)) \vdash_{,} (q', w, (\bar{y}_1, \ldots, \bar{y}_k, \epsilon, \ldots, \epsilon))$$

if

$$(q, aw, (y_1, \ldots, y_k, \epsilon, \ldots,)) \vdash (q', w, (\bar{y}_1, \ldots, \bar{y}_k, \epsilon, \ldots, \epsilon)).$$

Let $\vdash_{,}^{*}$ be the reflexive transitive closure of $\vdash_{,}$. Intuitively, $\vdash_{,}^{*}$ represents transitions in the $\mathcal{S}_1$-part of D and $\vdash_G^{*}$ in the $\mathcal{S}_2$-part of D. Since D is nested, if $C \vdash C'$, then either $C \vdash_{,} C'$ or $C \vdash_{G_D} C'$ but not both. Note that transitions $(p, a, (y_1, \ldots, y_k, \epsilon, \ldots, \epsilon)) \vdash (q', \epsilon, (y_1, \ldots, y_k, \epsilon, \ldots, \epsilon))$ occur as $\vdash_{,}$ and not as $\vdash_G$.

D is said to be in <u>factored form</u> (with <u>factor function</u> h) if h is a function from K_1 into $\mathcal{G}_D = \{G/G \subseteq G'_D\}$ such that

(1) if $(q_0, w, (\epsilon, \ldots, \epsilon)) \vdash^{*} (q, \epsilon, (y_1, \ldots, y_k, y'_1, \ldots, y'_\ell))$ then $h(q) \subseteq \{(\gamma_1, \ldots, \gamma_k)/\gamma_i \text{ in } \mathcal{S}_{\alpha_i}(y_i) \text{ for each } i\}$.

(2) if $m \geq 1$ and $(p, w, (y_1, \ldots, y_k, y'_1, \ldots, y'_\ell)) \vdash_G^{m} (q, \epsilon, (y_1, \ldots, y_k, y''_1, \ldots, y''_\ell))$, then $h(p) = h(q)$ and

$$(p, w, (y_1, \ldots, y_k, y'_1, \ldots, y'_\ell)) \vdash_{G \cap h(p)} (q, \epsilon, (y_1, \ldots, y_k, y''_1, \ldots, y''_\ell)).$$

The following two facts hold whenever D is in factored form with factor function h:

(1) If $(q_0, w, (\epsilon, \ldots, \epsilon)) \vdash^{*} (q, \epsilon, (y_1, \ldots, y_k, y'_1, \ldots, y'_\ell))$ and there

exists j_o such that $y'_{j_o} \neq \epsilon$, then $h(q) \neq \emptyset$. (For there exists a configuration C and m≥1 such that

$$(q_o, w, (\epsilon, \ldots, \epsilon)) \vdash^{*} C \vdash^{m}_{G_D} (q, \epsilon, (y_1, \ldots, y_k, y'_1, \ldots, y'_\ell)).$$

Hence $C \vdash^{m}_{G'_D \cap h(q)} (q, \epsilon, (y_1, \ldots, y_k, y'_1, \ldots, y'_\ell))$. By definition of $\vdash_{\overline{G'_D \cap h(q)}}$, $G'_D \cap h(q) \neq \emptyset$.)

(2) If $C_i \vdash_{G} C_{i+1}$ for $1 \leq i < m$, where each $C_i = (p_i, w_i, (y_1, \ldots, y_k, y'_{i1}, \ldots, y'_{i\ell}))$, then $\emptyset \neq h(p_1) = h(p_i)$, $1 \leq i \leq m$, and $C_i \vdash_{\overline{h(p_1)}} C_{i+1}$ for each $i<m$. In particular, $C_1 \vdash^{m-1}_{\overline{h(p_1)}} C_m$.

D is said to be in <u>restricted factored form</u> if it is in factored form and if $w \neq \epsilon$ whenever m≥1 and there exists a G such that

$$(q, w, (y_1, \ldots, y_k, \epsilon, \ldots, \epsilon)) \vdash^{m}_{G} (q', \epsilon, (y_1, \ldots, y_k, \epsilon, \ldots, \epsilon)).$$ [16]

To prove Lemma 4.3, we shall show that given D in $\mathcal{D}_3^N$, (a) there exists an equivalent device in $\mathcal{D}_3^N$ in restricted factored form, and (b) if D is in restricted factored form (and is quasi-realtime), then L(D) is in $\mathcal{L}(\mathcal{D}_1^N)_\sigma\, \mathcal{L}(\mathcal{D}_2^N)$ $(\mathcal{L}^t(\mathcal{D}_1^N)_\sigma\, \mathcal{L}^t(\mathcal{D}_2^N))$.

<u>Lemma A</u>. Given D in $\mathcal{D}_3^N$, there exists D′ in $\mathcal{D}_3^N$ such that

(a) D′ is in factored form,

(b) L(D) = L(D′),

and (c) D′ is quasi-realtime if D is quasi-realtime.

---

[16] From the definition of $\vdash_{G}$, m≥2.

<u>Proof</u>. Let $K_1' = K_1 \times \mathcal{B}_D \times \{1,2\}$. Let $D' = (K_1', \Sigma_1, \delta', (q_0, \emptyset, 1), F \times \{\emptyset\} \times \{1\}, \upsilon)$, where δ' is defined as follows (for arbitrary p in K_1, a in $\Sigma_1 \cup \{\epsilon\}$, and G in $\mathcal{B}$):

(1) $((p,\emptyset,1), (1(\alpha_1,\gamma_1), \ldots, 1(\alpha_k,\gamma_k), 1(\beta_1,\epsilon), \ldots, 1(\beta_\ell,\epsilon))$ is in $\delta'((p,G,2),\epsilon,(\gamma_1, \ldots, \gamma_k,\epsilon, \ldots,\epsilon))$ for all $G \neq \emptyset$ and $(\gamma_1, \ldots, \gamma_k)$ in G.

(2) $((p,G\cup\{(\gamma_1,\ldots, \gamma_k)\}, 1) (1(\alpha_1,\gamma_1), \ldots, 1(\alpha_k,\gamma_k), 1(\beta_1,\epsilon), \ldots, 1(\beta_\ell,\epsilon)))$ is in $\delta'((p,G,1),\epsilon,(\gamma_1, \ldots, \gamma_k,\epsilon, \ldots, \epsilon))$ for each $(\gamma_1, \ldots, \gamma_k)$ in $G_D'-G$.

(3) If $(q,(u_1, \ldots, u_k,u_1', \ldots, u_\ell'))$ is in $\delta(p,a,(\gamma_1, \ldots, \gamma_k,\epsilon,\ldots,\epsilon))$ and u_{j_0}' is not in $\psi_{\beta_{j_0}}(\epsilon)$ for some j_0, then $((q,G,2), (u_1, \ldots, u_k,u_1', \ldots, u_\ell'))$ is in $\delta'((p,G,1), a,(\gamma_1, \ldots, \gamma_k, \epsilon, \ldots, \epsilon))$ for all $(\gamma_1, \ldots, \gamma_k)$ in G.

(4) If $(q,(u_1, \ldots, u_k,u_1', \ldots, u_\ell'))$ is in $\delta(p,a,(\gamma_1,\ldots,\gamma_k,\gamma_1',\ldots,\gamma_\ell'))$, $(\gamma_1, \ldots, \gamma_k,\gamma_1', \ldots, \gamma_\ell')$ in G_D, and for some j_0, either u_{j_0}' is not in $\psi_{\beta_{j_0}}(\epsilon)$ or $\gamma_{j_0}' \neq \epsilon$, then $((q,G,2),(u_1, \ldots, u_k,u_1', \ldots, u_\ell'))$ is in $\delta'((p,G,2),a,$ $(\gamma_1, \ldots, \gamma_k,\gamma_1', \ldots, \gamma_\ell'))$ for all G containing $(\gamma_1, \ldots, \gamma_k)$.

(5) If $(q,(u_1, \ldots, u_k,u_1', \ldots, u_\ell'))$ is in $\delta(p,a,(\gamma_1, \ldots, \gamma_k,\epsilon,\ldots,\epsilon))$, $(\gamma_1, \ldots, \gamma_k,\epsilon, \ldots, \epsilon)$ in G_D and u_j' is in $\psi_{\beta_j}(\epsilon)$ for all j, $1 \leq j \leq \ell$, then $((q,\emptyset,1), (u_1, \ldots, u_k,u_1', \ldots, u_\ell'))$ is in $\delta'((p,\emptyset,1), a,$ $(\gamma_1, \ldots, \gamma_k,\epsilon, \ldots, \epsilon))$.

Obviously D' is in $\mathcal{D}_3^N$. Let h be the function defined by $h((p,G,i)) = G$ for all p in K_1, G in $\mathcal{B}$, and i in $\{1,2\}$. By inspection (since $G_D' = G_{D'}'$), D' is in factored form with factor function h, i.e., D' satisfies (a).

Consider (b). D' enters a state $(p,G,2)$ if and only if D has a $\vdash_G$ transition. Type 3 and type 4 rules imitate $\vdash_G$-transitions, and type 5 rules imitate $\vdash_{,}$ transitions in D. Since D is nested, every transition has an associated $\vdash_G$ or $\vdash_{,}$ transition. When the $\mathcal{S}_2$-tapes are ϵ, a type 1 rule can be used to go from a state $(p,G,2)$, where $h((p,G,1)) = G \neq \emptyset$, to a state $(p,\emptyset,1)$ and then enter the $\mathcal{S}_1$-part of the acceptor. Type 2 rules represent the guess that D executes a $\vdash_G$ transition. Once $h((p,G,1)) \neq \emptyset$, D' ultimately blocks or else executes (by a type 3 rule) at least one $\vdash_G$ transition. Thus $L(D) = L(D')$.

Consider (c). If $\#(G_D') = n_o$, then D' has at most $n+2$ consecutive ϵ-moves for each ϵ-move of D. Thus D' is quasi-realtime if D is quasi-realtime.

<u>Lemma B.</u> If D in $\mathcal{S}_3^N$ is in factored form, then there exists D'' in $\mathcal{S}_3^N$ such that

(a) D'' is in restricted factored form,

(b) $L(D'') = L(D)$,

and (c) D'' is quasi-realtime if D is quasi-realtime.

<u>Proof.</u> Let D be in factored form with factor function h. Since $(p,\epsilon,(x_1, \ldots, x_k,\epsilon, \ldots, \epsilon)) \overset{m}{\vdash_G} (q,\epsilon,(x_1, \ldots, x_k,\epsilon, \ldots, \epsilon))$, $m \geq 1$, implies $\emptyset \neq h(p) = h(q)$ and $(p,\epsilon,(x_1, \ldots, x_k,\epsilon, \ldots, \epsilon)) \overset{m}{\vdash_{h(p)}} (q,\epsilon,(x_1,\ldots,x_k,\epsilon,\ldots,\epsilon))$, we need only consider such transitions with $G = h(p) = h(q)$. Let S be the set of all (p,q) in $K_1 \times K_1$, where $p \neq q$ and $\emptyset \neq h(p) = h(q)$, such that there exist $\gamma_1, \ldots, \gamma_k$ satisfying

$$(p,\epsilon,(\gamma_1, \ldots, \gamma_k,\epsilon, \ldots, \epsilon)) \overset{*}{\vdash_{h(p)}} (q,\epsilon,(\gamma_1, \ldots, \gamma_k,\epsilon, \ldots, \epsilon)).$$

Let $K''_1 = K_1 \cup (K_1 \times (\Sigma_1 \cup \{\epsilon\}))$. Let $D'' = (K_1'',\Sigma_1,\delta'',q_o,F'',\upsilon)$, where $F'' = F \cup (F \times \{\epsilon\})$ and δ'' is defined as follows:

(1) $(q,(u_1, \ldots, u_k,u_1', \ldots, u_\ell'))$ is in $\delta''(p,a,(\gamma_1, \ldots, \gamma_k,\epsilon, \ldots, \epsilon))$

if $(q, u_1, \ldots, u_k, u_1', \ldots, u_\ell'))$ is in $\delta(p, a, (\gamma_1, \ldots, \gamma_k, \epsilon, \ldots, \epsilon))$, with u_j' in $\psi_{\beta_j}(\epsilon)$ for all j.

[By (1), the $\vdash^{*}_{\prime}$ transitions of D appear in D″.]

(2) $(q, (1(\alpha_1, \gamma_1), \ldots, 1(\alpha_k, \gamma_k), 1(\beta_1, \epsilon), \ldots, 1(\beta_\ell, \epsilon)))$ is in $\delta''(p, \epsilon, (\gamma_1, \ldots, \gamma_k, \epsilon, \ldots, \epsilon))$ for all (p,q) in S and all $(\gamma_1, \ldots, \gamma_k)$ in $h(p)$.

[Since $p \neq q$, the ϵ-move in (2) replaces at least two ϵ-moves in D. For if $(p, \epsilon, (y_1, \ldots, y_k, \epsilon, \ldots, \epsilon)) \vdash^{*}_{G} (q, \epsilon, (y_1, \ldots, y_k, \epsilon, \ldots, \epsilon))$ for $p \neq q$, then by definition of $\vdash_{G}$ at least two moves are needed.]

(3) If $(q, (u_1, \ldots, u_k, u_1', \ldots, u_\ell'))$ is in $\delta(p, a, (\gamma_1, \ldots, \gamma_k, \epsilon, \ldots, \epsilon))$, with $(\gamma_1, \ldots, \gamma_k)$ in $h(p) = h(q)$ and u_{j_0}' not in $\psi_{\beta_{j_0}}(\epsilon)$ for some j_0, then

(α) $((q, \epsilon), (u_1, \ldots, u_k, u_1', \ldots, u_\ell'))$ is in $\delta''(p, a, (\gamma_1, \ldots, \gamma_k, \epsilon, \ldots, \epsilon))$ if a is in Σ_1.

(β) $((q, b), (u_1, \ldots, u_k, u_1', \ldots, u_\ell'))$ is in $\delta''(p, b(\gamma_1, \ldots, \gamma_k, \epsilon, \ldots, \epsilon))$ for each b in Σ_1 if $a = \epsilon$.

[By (3), a sequence of $\vdash_{G}$ moves in D″ starts with a non-ϵ input, namely, either the input to D (3α), or a guess as to the eventual non-ϵ input symbol to be read by D (3β).]

(4) If $(q, (u_1, \ldots, u_k, u_1', \ldots, u_\ell'))$ is in $\delta(p, a, (\gamma_1, \ldots, \gamma_k, \gamma_1', \ldots, \gamma_\ell'))$, with $\gamma_{j_0}' \neq \epsilon$ for some j_0 and $(\gamma_1, \ldots, \gamma_k)$ in $h(p) = h(q)$, then

(α) $((q, \epsilon), (u_1, \ldots, u_k, u_1', \ldots, u_\ell'))$ is in $\delta''((p, \epsilon), a, (\gamma_1, \ldots, \gamma_k, \gamma_1', \ldots, \gamma_\ell'))$.

(β) $((q,\epsilon),(u_1, \ldots, u_k,u_1', \ldots, u_\ell'))$ is in $\delta''((p,a),\epsilon,(\gamma_1, \ldots, \gamma_k, \gamma_1', \ldots, \gamma_\ell'))$ if a is in Σ_1.

(γ) $((q,b),(u_1, \ldots, u_k,u_1', \ldots, u_\ell'))$ is in $\delta''((p,b),\epsilon(\gamma_1, \ldots, \gamma_k, \gamma_1', \ldots, \gamma_\ell'))$ for all b in Σ_1 if $a=\epsilon$.

[By (4), D'' can enter no state (q,ϵ) until the guess as to the non-ϵ input of D has been verified. If D'' enters a configuration $((q,a),w,(y_1,\ldots,y_k,\epsilon,\ldots,\epsilon))$ with $a\neq\epsilon$, then D'' blocks since it has traced out a computation on ϵ-input handled by a type 2 rule. Otherwise, a $\overset{*}{\underset{G}{\vdash}}$ transition is unchanged.]

(5) $(p,(1(\alpha_1,\gamma_1), \ldots, 1(\alpha_n,\gamma_n),1(\beta_1,\epsilon), \ldots, 1(\beta_n,\epsilon)))$ is in $\delta''((p,\epsilon),\epsilon,(\gamma_1, \ldots, \gamma_k,\epsilon, \ldots, \epsilon))$ for each p in K_1 and $(\gamma_1, \ldots, \gamma_k)$ in $h(p)$. [After tracing a $\overset{*}{\underset{G}{\vdash}}$ computation on non-ϵ input, D'' returns to a K_1-state and imitates either $\overset{*}{\underset{\prime}{\vdash}}$ or $\overset{*}{\underset{G}{\vdash}}$ transitions.]

Since only type 5 rules add ϵ-rules not simulating ϵ-rules in D and these cannot be applied twice in a row, D'' is quasi-realtime if D is.

The only transitions of D not represented in D'' are $(p,\epsilon,(y_1, \ldots, y_k,\epsilon, \ldots, \epsilon))\overset{*}{\underset{h(p)}{\vdash}} (q,\epsilon,(y_1, \ldots, y_k,\epsilon, \ldots, \epsilon))$ and these are covered by (2). The only new transitions are (2) and (5). Thus $L(D)= L(D'')$.

Let $\bar{h}$ be the function on K_1'' defined by $\bar{h}(p) = \bar{h}((p,a)) = h(p)$ for each p in K_1 and a in $\Sigma_1\cup\{\epsilon\}$. Then D'' is in factored form with factor function $\bar{h}$. Now $\overset{*}{\underset{G}{\vdash}}$ computations in D'' start with a type 3 rule. Since all type 3 rules have non-ϵ input, D'' is in restricted factored form.

We are now ready for Lemma 4.3.

<u>Lemma 4.3</u>. $\mathcal{L}^t(\mathcal{D}_3^N) \subseteq \mathcal{L}^t(\mathcal{D}_1^N)\sigma\ \mathcal{L}^t(\mathcal{D}_2^N)$

and $\mathcal{L}(\mathcal{D}_3^N) \subseteq \mathcal{L}(\mathcal{D}_1^N)\ \sigma\ \mathcal{L}(\mathcal{D}_2^N)$.

<u>Proof</u>. Let $D = (K_1,\Sigma_1,\delta,q_o,F,\nu)$ be in $\mathcal{D}_3^N$. Suppose each α in ν is in $\mathcal{C}_1$. Then D is in $\mathcal{D}_1^N$ and $L(D) = \tau(L(D))$, where τ is the ϵ-free substitution defined by $\tau(a) = \{a\}$ for each a in Σ_1. Since the AFL $\mathcal{L}^t(\mathcal{D}_2^N)$ contains each ϵ-free regular set and thus each $\{a\}$, each $\tau(a)$ is in $\mathcal{L}^t(\mathcal{D}_2^N)$, whence the result. Similarly, if all α in ν are in $\mathcal{C}_2$, then D is in $\mathcal{D}_2^N$ and $L(D) = \tau(\{a\})$, whence $\tau(a) = L(D)$ and the regular set $\{a\}$ is in $\mathcal{L}^t(\mathcal{D}_1^N)$. Thus assume that $\nu = (\alpha_1, \ldots, \alpha_k,\beta_1, \ldots, \beta_\ell)$, with k, $\ell\geq 1$, $\{\alpha_1, \ldots, \alpha_k\} \subseteq \mathcal{C}_1$ and $\{\beta_1, \ldots, \beta_\ell\}\subseteq\mathcal{C}_2$. By Lemmas A and B we may assume that D is in restricted factored form with factor function h. We shall say that G <u>describes</u> $(y_1, \ldots, y_k)$ if $G \subseteq \{(\gamma_1,\ldots,\gamma_k)/\gamma_i \text{ in } g_{\alpha_i}(y_i) \text{ for each } i\}$. We shall say $(p,(y_1, \ldots, y_k))$ is <u>accessible</u> if $(q_o,w,(\epsilon, \ldots, \epsilon))\overset{*}{\vdash}(p,\epsilon,(y_1, \ldots, y_k,y_1', \ldots, y_\ell'))$ for some w and some $(y_1', \ldots, y_\ell')$. Without loss of generality we may assume that if p is in K_1, then there exists $(y_1, \ldots, y_k)$ such that $(p,(y_1, \ldots, y_k))$ is accessible.

Since D is in factored form, if

$(p,w,(y_1, \ldots, y_k,y_1', \ldots, y_\ell'))\overset{m}{\underset{G}{\vdash}} (q,\epsilon,(y_1, \ldots, y_k,y_1'', \ldots, y_\ell''))$ for

some $m\geq 1$ and if $(p,(y_1, \ldots, y_k))$ is accessible, then the following hold:

(1) $h(p) = h(q)$ and $h(p)$ describes $(y_1, \ldots, y_k)$.

(2) $(p,w,(y_1, \ldots, y_k,y_1', \ldots, y_\ell'))\overset{m}{\underset{h(p)}{\vdash}}(q,\epsilon,(y_1,\ldots,y_k,y_1'',\ldots,y_\ell''))$.

(3) If h(p) describes $(\bar{y}_1, \ldots, \bar{y}_k)$, then

$$(p,w,(\bar{y}_1, \ldots, \bar{y}_k, y'_1, \ldots, y'_\ell)) \vdash^{m}_{h(p)} (q,\epsilon,(\bar{y}_1, \ldots, \bar{y}_k, y''_1, \ldots, y''_\ell)).$$

For each (p,q) in $K_1 \times K_1$ with $h(p) = h(q) \neq \emptyset$, let <p,q> be a new symbol and let

$$L_{p,q} = \{w/(p,w,(y_1, \ldots, y_k,\epsilon, \ldots, \epsilon)) \vdash^{*m}_{h(p)} (q,\epsilon,(y_1, \ldots, y_k,\epsilon, \ldots, \epsilon))$$
$$\text{for some } m \geq 1 \text{ and some } y_1, \ldots, y_k\}.$$

(The condition $m \geq 1$ is needed for the case p=q since otherwise $L_{p,q}$ might trivially contain ϵ.) Since D is in restricted factored form, $L_{p,q}$ is obviously ϵ-free. Let $K_{p,q} = \{s \text{ in } K_1/h(s) = h(p)\}$ and let $D_{p,q} = (K_{p,q}, \delta_{p,q}, p, \{q\}, (\beta_1, \ldots, \beta_\ell))$, where $\delta_{p,q}$ is defined as follows (for all appropriate s_1, s_2, a, etc): If $(s_2,(u_1, \ldots, u_k, u'_1, \ldots, u'_\ell))$ is in $\delta(s_1,a,(\gamma_1, \ldots, \gamma_k, \gamma'_1, \ldots, \gamma'_\ell))$, with $h(s_1) = h(s_2) = h(p)$ and there exist j_0 such that either u'_{j_0} is not in $\psi_{\beta_{j_0}}(\gamma'_{j_0})$ or $\gamma'_{j_0} \neq \epsilon$; then $(s_2,(u'_1, \ldots, u'_\ell))$ is in $\delta_{p,q}(s_1,a,(\gamma'_1, \ldots, \gamma'_\ell))$.

Suppose $h(p) = h(q) \neq \emptyset$. Obviously $D_{p,q}$ is quasi-realtime if D is quasi-realtime. If

$$(p,w,(y_1, \ldots, y_k,\epsilon, \ldots, \epsilon)) \vdash^{*}_{h(p)} (s_1,\epsilon,(y_1, \ldots, y_k, y'_1, \ldots, y'_\ell))$$

and $(s_1,a,(y_1, \ldots, y_k, y'_1, \ldots, y'_\ell)) \vdash_{h(p)} (s_2,\epsilon,(y_1, \ldots, y_k, y''_1, \ldots, y''_\ell))$,

then $h(p) = h(s_1) = h(s_2)$ and there exists j_0 such that either $y'_{j_0} \neq \epsilon$ or $y''_{j_0} \neq \epsilon$, so that

$$(s_1,a,(y'_1, \ldots, y'_\ell)) \vdash_{D_{p,q}} (s_2,\epsilon,(y''_1, \ldots, y''_\ell)).$$

Hence $L_{p,q} \subseteq L(D_{p,q})$. On the other hand, suppose $(s_1, a, (y_1', \ldots, y_\ell'))$ $\vdash_{D_{p,q}}$ $(s_2, \epsilon, (y_1'', \ldots, y_\ell''))$. Then $h(p) = h(s_1) = h(s_2)$, there exists $(y_1, \ldots, y_k)$ such that $h(p)$ describes $(y_1, \ldots, y_k)$ (by (1)), and there exists j_o such that either $y_{j_o}' \neq \epsilon$ or $y_{j_o}'' \neq \epsilon$. Thus

$$(s_1, a, (y_1, \ldots, y_k, y_1', \ldots, y_\ell')) \vdash_{h(p)} (s_2, \epsilon, (y_1, \ldots, y_k, y_1'', \ldots, y_\ell'')).$$

Hence $L(D_{p,q}) \subseteq L_{p,q}$, so that $L(D_{p,q}) = L_{p,q}$.

Let Σ_2 be the set of all $<p,q>$. Let τ be the substitution on $\Sigma_1 \cup \Sigma_2$ defined by $\tau(a) = \{a\}$ for each a in Σ_1 and $\tau(<p,q>) = L_{p,q}$ for each $<p,q>$ in Σ_2. Then τ is an ϵ-free substitution by $\mathcal{L}(\mathcal{D}_2)$, and if D is quasi-realtime then τ is an ϵ-free substitution by $\mathcal{L}^t(\mathcal{D}_2)$.

Now let $\overline{D} = (K_1, \Sigma_1 \cup \Sigma_2, \overline{\delta}, q_o, F, (\alpha_1, \ldots, \alpha_k))$, where $\overline{\delta}$ is defined as follows:

(4) $(q, (u_1, \ldots, u_k))$ is in $\overline{\delta}(p, a, (\gamma_1, \ldots, \gamma_k))$ if $(q, (u_1, \ldots, u_k, u_1', \ldots, u_\ell'))$ is in $\delta(p, a, (\gamma_1, \ldots, \gamma_k, \epsilon, \ldots, \epsilon))$ and u_j' is in $\psi_{\beta_j}(\epsilon)$ for all j.

(5) $(q, (l(\alpha_1, \gamma_1), \ldots, l(\alpha_k, \gamma_k)))$ is in $\overline{\delta}(p, <p,q>, (\gamma_1, \ldots, \gamma_k))$ for all p, q such that $h(p) = h(q) \neq \emptyset$ and $(\gamma_1, \ldots, \gamma_k)$ is in $h(p)$.

If $(p, a, (y_1, \ldots, y_k)) \vdash_{\overline{D}} (q, \epsilon, (\overline{y}_1, \ldots, \overline{y}_k))$, with a in $\Sigma_1 \cup \{\epsilon\}$, then $\tau(a) = \{a\}$ and $(p, a, (y_1, \ldots, y_k, \epsilon, \ldots, \epsilon)) \vdash_D (q, \epsilon, (\overline{y}_1, \ldots, \overline{y}_k, \epsilon, \ldots, \epsilon))$. Thus $(q, (\overline{y}_1, \ldots, \overline{y}_k))$ is accessible if $(p, (y_1, \ldots, y_k))$ is. If

$(p,b,(y_1, \ldots, y_k)) \vdash_{\overline{D}} (q,\epsilon,(\overline{y}_1, \ldots, \overline{y}_k))$, with b in Σ_2, then $b = \langle p,q \rangle$, $\tau(b) = L_{p,q}$, and $(\overline{y}_1, \ldots, \overline{y}_k) = (y_1, \ldots, y_k)$. In addition, if w is in $L_{p,q}$ and h(p) describes $(y_1, \ldots, y_k)$, then

$$(p,w,(y_1, \ldots, y_k,\epsilon, \ldots, \epsilon)) \overset{*}{\vdash}_{\overline{D},h(p)} (q,\epsilon,(y_1, \ldots, y_k,\epsilon, \ldots, \epsilon)).$$

(For if a transition in D holds for some $(y_1, \ldots, y_k)$ described by h(p), it holds for all such $(y_1, \ldots, y_k)$.) From this it readily follows that $L(D) = \tau(L(\overline{D}))$. Clearly $\overline{D}$ is in $\mathcal{D}_1$ and $\overline{D}$ is quasi-realtime if D is. Hence L(D) is in $\mathcal{L}(\mathcal{D}_1)\ \sigma\ \mathcal{L}(\mathcal{D}_2)$, and if D is quasi-realtime then L(D) is in $\mathcal{L}^t(\mathcal{D}_1)\ \sigma\ \mathcal{L}^t(\mathcal{D}_2)$.

BIBLIOGRAPHY

[1] Aho, A.V., "Nested Stack Automata," to appear in the Journal of the Association for Computing Machinery.

[2] Chomsky, N. and M.P. Schutzenberger, "The Algebraic Theory of Context-Free Languages, in P. Braffort and D. Hirschberg (eds.) Computer-Programming and Formal Systems. Amsterdam: North Holland Publishing Co., 1963, pp. 118-161.

[3] Fischer, P.C., "Turing Machines with Restricted Memory Access," Information and Control, Vol. 9 (1966), pp. 364-379.

[4] Fischer, P.C., A.K. Meyer and A.L. Rosenberg, "Counter Machines and Counter Languages," Mathematical Systems Theory, Vol. 2 (1968), pp. 265-283.

[5] Ginsburg, S. and S.A. Greibach, "Abstract Families of Languages," to appear in a memoir of the American Mathematical Society.

[6] Ginsburg, S., S.A. Greibach and M. Harrison, "One-Way Stack Automata," Journal of the Association for Computing Machinery, Vol. 14 (1967), pp. 389-418.

[7] Ginsburg, S., S.A. Greibach and J. Hopcroft, "Pre-AFL," to appear in a memoir of the American Mathematical Society.

[8] Ginsburg, S. and M.A. Harrison, "One-Way Nondeterministic Real-Time List-Storage Languages," Journal of the Association for Computing Machinery, Vol. 15 (1968), pp. 428-446.

[9] Ginsburg, S. and J. Hopcroft, "Images of AFL Under Certain Families of Homomorphisms," (In Preparation).

[10] Ginsburg, S. and E.H. Spanier, "Finite-Turn Pushdown Automata," SIAM Journal of Control, Vol. 4 (1966), pp. 429-453.

[11] Ginsburg, S. and E.H. Spanier, "Control Sets on Grammars," Mathematical Systems Theory, Vol. 2 (1968), pp. 159-177.

[12] Ginsburg, S. and E.H. Spanier, "Derivation-Bounded Languages," Journal of Computer and System Sciences, Vol. 2 (1968), pp. 228-250.

[13] Ginsburg, S. and E.H. Spanier, "Substitutions in Families of Languages," SDC document TM-738/049/00, September 1968.

[14] Greibach, S.A., "A New Normal-Form Theorem for Context-Free Phrase Structure Grammars," Journal of the Association for Computer Machinery, Vol. 12 (1965), pp. 42-52.

[15] Greibach, S.A., "An Infinite Hierarchy of Context-Free Languages," Journal of the Association for Computer Machinery, Vol. 16 (1969), pp. 91-106.

[16] Greibach, S.A., "Checking Automata and One-Way Stack Languages," SDC document TM-738/045/00, April 1968.

[17] Greibach, S.A., "Chains of Full AFL's," (In Preparation.)

[18] Hopcroft, J. and J. Ullman, "An Approach to a Unified Theory of Automata," Bell System Technical Journal, Vol. 46 (1967), pp. 1793-1829.

[19] Minsky, M., "Recursive Unsolvability of Post's Problem of "Tag" and Other Topics in Theory of Turing Machines," Annals of Mathematics, Vol. 74 (1961), pp. 437-455.

[20] Nivat, M., "Transductions des languages de Chomsky," unpublished thesis Grenoble University, 1967.

[21] Yntema, M.K., "Inclusion Relations Among Families of Context-Free Languages," Information and Control, Vol. 10 (1967), pp. 572-597.

Unclassified
Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) System Development Corporation 2500 Colorado Avenue Santa Monica, California 90406	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP

3. REPORT TITLE

MULTITAPE AFA.

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

5. AUTHOR(S) (First name, middle initial, last name)

Sheila Greibach
Seymour Ginsburg

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
2 January 1969	67	21

8a. CONTRACT OR GRANT NO. F1962867C0008, Grant No. AF-AFOSR-1203-67A Project, Task, Work Unit Nos. 5632-05-01 c. DoD Element 6144501F d. DoD Subelement 681305	9a. ORIGINATOR'S REPORT NUMBER(S) TM-738/050/00 Scientific report no. 23 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) AFCRL-69-0173

10. DISTRIBUTION STATEMENT

Distribution of this document is unlimited. It may be released to the Clearinghouse, Department of Commerce, for sale to the general public.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
Air Force Office of Scientific Research Office of Aerospace Research, USAF Arlington, Virginia	Air Force Cambridge Research Laboratories (CRB) L. G. Hanscom Field Bedford, Massachusetts 01730

13. ABSTRACT

The present paper gives device representations, via multitape AFA, for the families of languages which result from applying the $\wedge$ and the substitution operations to AFL. In particular, if $\mathcal{D}_1$ and $\mathcal{D}_2$ are multitape AFA (i.e., certain families of multi-storage tape acceptors), then $\mathcal{D}_1 \wedge \mathcal{D}_2$ is defined as the family of multitape acceptors which results when the tapes of $\mathcal{D}_1$ and $\mathcal{D}_2$ are coalesced, with the $\mathcal{D}_1$- tapes preceding those in $\mathcal{D}_2$. It is shown that the smallest full AFL containing $\mathcal{L}(\mathcal{D}_1) \wedge \mathcal{L}(\mathcal{D}_2) = \{L_1 \cap L_2 / L_i \text{ in } \mathcal{L}(\mathcal{D}_i)\}$ is $\mathcal{L}(\mathcal{D}_1 \wedge \mathcal{D}_2)$. For each multitape AFA $\mathcal{D}$, a set $\mathcal{D}^N$ of "nested" multitape acceptors is defined. It is shown that if $\mathcal{D}_1$ and $\mathcal{D}_2$ are single-tape AFA, then the family of languages obtained from $(\mathcal{D}_1 \wedge \mathcal{D}_2)^N$ is the family of languages obtained by substituting the AFL defined by $\mathcal{D}_2$ into the AFL defined by $\mathcal{D}_1$.

DD FORM 1 NOV 65 1473

Unclassified
Security Classification

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Families of languages						
Abstract family of languages (AFL)						
AFL						
Abstract family of multitape acceptors (AFA)						
AFA						